A Comprehensive Survey of Blockchain: from Theory to IoT Applications and Beyond

Mingli Wu, Kun Wang, *Senior Member, IEEE,* Xiaoqin Cai, Song Guo, *Senior Member, IEEE,* Minyi Guo, *Fellow, IEEE* and Chunming Rong, *Senior Member, IEEE*

***Abstract*—As an innovated and revolutionized technology, blockchain has been applied in many fields, such as cryptocurrency, food traceability, identity management, or even market prediction. To discover its great potential, both the industry and academia have paid great attention to it and numerous researches have been conducted. Based on the literatures and industry whitepapers, in this survey, we unroll and structure the blockchain related discoveries and scientific results in many aspects. Particularly, we classify blockchain technologies into four layers and carry out a comprehensive study on the consensus strategies, the network, and the applications of blockchain. Different blockchain applications are put into the corresponding categories based on the fields, especially in Internet of Things (IoT). When introducing each layer, we not only organize and summarize the related works, but also discuss the fundamental issues and future research directions. We hope this survey could shed some light on the research of blockchain and serve as a guide for further studies.**

***Index Terms*—Blockchain, IoT, consensus algorithms, P2P Network, anonymity, scalability, attacks, extension, application**

I. Introduction

LAST decade has witnessed the rise and prosperity of the blockchain technology, which has been deployed in various fields. According to the definition of National Institute of Standards and Technology (NIST), blockchain is an immutable digital ledger system implemented in a distributed fashion (i.e., without a central repository) and usually without a central authority [1]. It was proposed by Nakamoto Satoshi [2] in 2008 to introduce bitcoin, a kind of digital currency that has also been prevailing for recent years. With the aim of achieving complete decentralization, Nakamoto first introduces a consensus mechanism named Proof of Work (PoW), in which the miners are incentivized to mine the blocks by competing their hash power. Literally, in PoW, more work the miners do, higher chances the miner mines the block and get the tokens (e.g., bitcoin in the Bitcoin).

Though blockchain is an innovative technology, it is not a new invention. Technically, it integrates many various existing techniques, such as cryptography, consensus in distributed systems, game theory, network security, etc. However, it does bring multitudes of new research challenges and issues. Due to its wide coverage, researchers from different background have done numerous works to enhance and promote the blockchain technology. For instance, the blockchain consensus mechanism is one of the most fundamental techniques that evolve from the early PoW to other derivatives, such as Proof of Stake (PoS) [3], Delegated Proof of Stake (DPoS) [4], Proof of Burn (PoB) [5], etc. Different consensus strategies are proposed from different considerations. For instance, considering that PoW consumes too much energy, PoS is proposed. Given the transaction speed issue, DPoS is proposed later after PoS.

Networking is another fundamental technique that plays a critical role in blockchain. Blockchain is far more than blocks chaining with hash functions. It is a distributed public ledger that records transactions via a peer-to-peer network. Every node is supposed to maintain a ledger so that blockchain gains the property of non-repudiation. However, maintaining a peer-to-peer (P2P) network with thousands of nodes is a non-trivial work. One obvious concern with blockchain is the scalability problem, which has been widely discussed. Bitcoin only supports 7 transactions per second (TPS) and even takes 10 minutes for confirming one transaction. For Ethereum, it also only supports 15 TPS. The scalability issue has become a big hindrance to the development of blockchain. Due to its openness, this network also suffers from the attacks in analogy with other networks. In blockchain networks, a replay attacker can maliciously repeat valid transactions to gain benefits especially when a blockchain is hard-forked. A fork occurs when two or more longest subchains coincide with the same length. Other attacks, such as Sybil attack [6], DDoS attacks, have been widely discussed [7] [8].

Based on the fundamental consensus mechanisms and P2P maintaining network, there are technical extensions aiming to promote the traditional blockchain with functionality and scalability. A smart contract is a distinctive blockchain extension being widely deployed. It is a digital contract keeping in blockchain and maintained by the blockchain network. Statistics indicate there are 19,366 contracts in Ethereum networks [9]. The number of Github projects related with smart contract has been 445 units [10]. Smart contract can be utilized for various applications, such as thwarting DDoS attack [11], building DApp [12], designing voting protocol [13], etc. The security issues of smart contract have also been discussed in many works [9] [14]. Similar to smart contract, sidechain is another blockchain extension that enlarges the functionality of

M. Wu and S. Guo are with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China (e-mail: mingliwu55@gmail.com, song.guo@polyu.edu.hk).

K. Wang is with the Department of Electrical and Computer Engineering, University of California, Los Angeles, 90095, CA, USA (e-mail: wangk@ucla.edu).

X. Cai and M. Guo are with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: guo-my@cs.sjtu.edu.cn, cai-xq@sjtu.edu.cn).

C. Rong is with the Center for IP-based Service Innovation, the University of Stavanger, Stavanger, Norway (e-mail: chunming.rong@uis.no).

blockchain. Sidechain could enable digital assets transferred between the sidechains and main chain [15]. Before applying sidechain, some factors need to be considered first, such as the mining power coordination [16].

In addition to the extension created to support blockchain applications, blockchain has been widely deployed in numerous fields. One prominent application of blockchain is cryptocurrency. Statistics indicate there are over 1,565 kinds of cryptocurrencies and still growing [17]. Though the market has been inundated with this cryptocurrency fever, the application of blockchain is far beyond cryptocurrency. Because of its innovativeness, blockchain has also been probed to be widely applied, such as medical record management [18], supply chain [19], certificate audit [20], privacy protection [21], etc. Blockchain4EU [22] is a project aiming to identify blockchain uses and impacts in various applications from assets monitoring, intellectual property rights to authentication.

Considering the fast development of the above-mentioned blockchain techniques and blockchain's worldwide impacts, some works have made their attempts to summarize blockchain literature in different aspects. Tschorsch and Scheuermann [23] conducted a technical survey on decentralized digital currencies rather than emphasize blockchain technology. Holub and Johnson [24] searched 1,206 papers on Bitcoin and category them into six disciplines. Baliga [25] provided an overview of the consensus algorithms adapted by famous blockchain systems. Atzei et al. [26] surveyed the attacks against smart contract and provide a taxonomy of common smart contract programming pitfalls. Anh et al. [27] gave their focus on private blockchain in a data processing view. However, although blockchain has been prevailing for several years, there is no comprehensive study to shed a light on the research and application of blockchain. Considering this, we conduct this survey to bridge this gap. It is to be mentioned that NIST also conducts an overview [1] of blockchain technology but from a technical perspective, rather than both technical and research angle of this paper.

Particularly, we summarize the contributions we make in this paper as follows:

- In analogy with TCP/IP four-layer Internet protocol suite, we propose a four-layer blockchain architecture, which is able to conclude all the related works about blockchain.
- In different parts (i.e., the cryptographic, consensus mechanism, network, application) of this paper, we compare numerous works and analyze their strengths and weaknesses to serve as guides for further researches.
- We make some suggestions to facilitate the development of blockchain and point out several promising research directions of blockchain technology.

The remainder of this paper is structured as follows. In section II, we introduce the background of blockchain, including the cryptography elements, the basic data structures of blocks and blockchain, the mining process and the blockchain architecture. In section III, we introduce the blockchain peer-to-peer network and the multi-chain network. We also describe the transaction process and discuss the scalability of blockchain. Attacks against blockchain networks and corresponding countermeasures are analyzed at the end of this section. In section IV, the fundamental consensus mechanisms of blockchain are presented. We investigate the popular blockchain consensus algorithms, compare their applications and point out their strengths and weaknesses. Attacks against the consensus algorithms are also discussed in this part. In section V, we introduce the various blockchain applications and categorize them into different fields. In section VI, we discuss the blockchain challenges and envision the opportunities. Finally, we conclude this survey in section VII.

II. Background

A. Overview

A financial system aiming to do transactions (or transfer money) in an orderly manner requires a ledger to record the transaction history and verify each transaction strictly. In history, bookkeeping can be dated back to 1494, when the double-entry bookkeeping system was first developed by the Italian mathematician Lucca Pacioli. The double-entry bookkeeping method records the source and destination of each account simultaneously, and introduces account verification into the accounting process for the first time, improving the reliability of accounting [28]. Nowadays, it is the financial companies who are responsible for their own ledgers. For instance, Alipay records its own centralized ledger. Blockchain technology develops from version 1.0, 2.0 to 3.0, which are respectively represented by Bitcoin, smart contract, and decentralized applications (DApps). In blockchain 1.0, mounts of cryptocurrencies are developed, such as Bitcoin, Eth, and XRP, which flourish the digital asset transaction market and make blockchain known to the public. In blockchain 2.0, a smart contract that is a digital contract can be triggered by some predefined conditions attracts people's attention. Ethereum plays a leading role in promoting the smart contract conception. By utilizing smart contracts, the transactions on blockchain can be more convenient. In blockchain 3.0, lots of DApps are developed to improve the usability of blockchain. In section V, we will further explain the three blockchain ages.

Different from the traditional centralized bookkeeping methods, blockchain is proposed as a decentralized and distributed ledger. In blockchain network, every peer node processes a complete copy of the ledger, making tampering becomes nearly impossible. The concept of blockchain first appeared in Nakamoto's work titled as "*Bitcoin: A Peer-to-Peer Electronic Cash System*" [2]. There is no clear definition of the blockchain in this paper and it is simply described as the accounting records for bitcoin transactions. However, we can still clearly understand what exactly it is according to its operation process. Concisely, a blockchain is a decentralized, distributed and public digital ledger that is used to record transactions across many computers [29].

The reason why blockchain gets continuous attention from the public is that it achieves the characteristics of decentralization, trust, transparency, anonymity, and non-repudiation. First, the distributed network architecture achieves decentralization, and the ledger is maintained by numerous nodes of the entire

network. Second, trust can be achieved without a costful third party of authority. In blockchain, the acquisition of trust is guaranteed by consensus mechanisms. Third, since the transactions in blockchain are public, every transaction can be fully recognized and the transaction process is also visible, thus achieving transparency. Fourth, anonymity is attempted to be attained through pseudonyms (i.e., changing transaction addresses). It needs to be mentioned that exploiting pseudonyms can only achieve "pseudoanonymity" [30] instead of perfect or true anonymity because of multi-input issue [2] and other deanonymization methods [31] [32]. Also, once a block is appended to the blockchain, it cannot be revoked again. The blocks can hardly be modified in this case, thus achieving non-repudiation, which is the final characteristic of blockchain.

B. Block and Blockchain

A block is a data structure recording the transaction records. Specifically, it is made up of a block header and many transaction records. Take bitcoin blockchain structure as an example, the block header contains the hash value of its previous block, the time stamp, Merkle root, a nonce, and other information. Because of the previous hash value, each block can be chained together, thus creating a blockchain. The bitcoin blockchain structure is shown in Fig. 1. It is to be mentioned that other blockchains also show a similar structure. The block hash can be used to identify each block, and the previous block header hash ensures that no previous block can be changed without changing this block's header. In another word, if an adversary attempts to tamper with one block, he also needs to tamper with all the later existing blocks. In Bitcoin, the transaction records in each block are organized into a Merkle tree. A Merkle tree is a hash tree in which every leaf node is labeled with the hash of a data block and every non-leaf node is labeled with the cryptographic hash of the labels of its child nodes [33]. The Merckle tree is depicted in Fig. 2. The Merkle root in block header is derived from the hashes of all the transactions included in the block. Therefore, no transactions can be modified without modifying the header.

C. Block Mining and Forks

Mining is a process that creates blocks. In Bitcoin, miners have to compete their hash powers to create the blocks by solving the puzzles. However, not all blockchains need mining, such as Ripple [34], a cryptocurrency akin to bitcoin. Whether it needs block mining depends on the blockchain design. In blockchains that need mining, the blocks are created in a certain frequency. The blocks in Ethereum are produced between 14 and 15 seconds, while it needs 10 minutes in bitcoin. The reason why it needs mining is to keep the blockchain decentralized. Individuals or groups from all the world join the mining forces to keep the blockchain as a decentralized design without a central authority. The miners are incentivized to mine the blocks for tokens. Different blockchains take different mining strategies, and the advantages and disadvantages of these strategies will be discussed in section IV. We will take bitcoin as an example to explain the block mining process. In Bitcoin, if a miner successfully mines a new valid block, he or she will be rewarded for some bitcoins, which worth some real-world currency according to the cryptocurrency transaction market conversion rate. To mine a valid block, the miner in bitcoin should solve a cryptographic puzzle by finding a SHA-256 hash value less than a predefined target value. The difficulty of solving the computational puzzle is calibrated after 2016 blocks having been mined to keep the block mining time as 10 minutes. To solve the puzzle, the miner has to do brute force calculation, making the block mining as a computation competition between the miners. Bitcoin takes PoW to quantify the miner's ability to solve the puzzle. In practice, miners intend to collaborate into pools [35] to gain more computation advantages, in which the pool members share the rewards.

Once a valid block is mined, the miner will broadcast the block to the public. After the validation of the peer nodes, this new block will be appended to the blockchain. Blockchain follows the rules of the longest chain (i.e., the chain with the longest length is the valid path to follow). However, due to the distributed nature and the network delay, it is possible that two or more divergent paths with the same longest path are found simultaneously, resulting in forks in blockchain. When the forks occur, the miners are free to follow one of the temporal longest paths. When one fork is extended to be the longest chain, all the miner will come to a consensus and follow it instead of other forks, leaving the other previous valid blocks being abandoned as *orphaned* blocks. Though the forks can be solved by the longest chain constraint, there are still forks in reality for various reasons. Generally, the forks can be classified into two categories: hard fork and soft fork.

- Hard Fork. When a protocol change occurs so that the software validating according to the old rules regards the blocks produced according to the new rules as invalid, then a hard fork is required [36]. If a miner updates to the new protocol, he may create new blocks that are not compatible with the other peers who still obey the old rules. The protocol change can be adding new features to enhance the network functionality or simply increasing the block size. A typical example of hard fork is Ethereum Classic blockchain [37], which is hard forked from the original Ethereum chain to remedy the losses caused after the Decentralized Autonomous Organization (DAO) was hacked in 2016. The recent news goes that the Ethereum community is considering a hard fork to limit the effectiveness of ASIC (Application Specific Integrated Circuit) after Bitmain announced an ASIC miner for an "ASIC-Proof" Ethereum [38]. An example of hard fork is as shown in Fig. 3
- Soft Fork. In contrast to hard fork, a soft fork is a change of rules that is backward compatible (i.e., blocks or transactions are still recognized as valid by the old software). Miners or users can keep running the old software after a soft fork. Also, they can still be part of the same network when the users have updated to the new version. Since the new version blocks are accepted by old version miners but the updated miners accept only the new version blocks, the chain with the new version

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2019.2922538, IEEE Internet of Things Journal

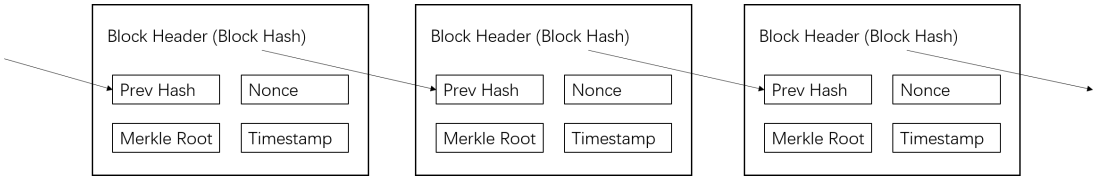


Fig. 1. A simplified blockchain.

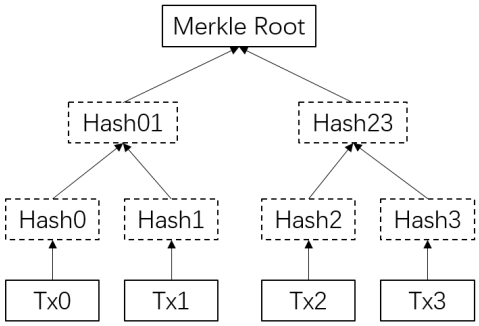


Fig. 2. A Merkle tree.

blocks will gradually become the longest one as more miners update. An example of soft fork is as shown in Fig. 4. The soft fork requires only the majority of miners to upgrade, which makes it less disruptive and safer than hard fork. One example of soft fork is software update of BIP 66 [39], which aims to improve the security of Bitcoin transactions.

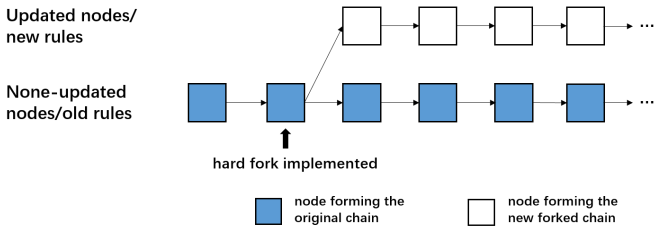


Fig. 3. Hard fork

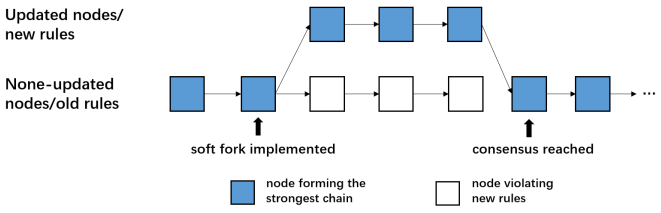


Fig. 4. Soft fork

D. Transactions

Transactions need to be recorded in ledgers. In blockchain, each transaction is an open record. In general, the transaction records the number of inputs and outputs as well as the address of assignments. Take an example of transactions in Bitcoin, the input coming from an unused output is called UTXO (Unspent Transaction Output), which is considered as a basic unit of transaction records. UTXO cannot be split unless adjusting the input and output to complete the specified transaction. For example, if UTXO is less than the target, one can accumulate multiple UTXOs as inputs; if UTXO is greater than the target, one can trade your address as a change-of-zero output. Each transaction will consume the existing UTXO and generate a new UTXO. The value of the transfer is accomplished through UTXO changes.

The access and consumption of UTXO are guaranteed by digital signatures. Only the correct digital signatures are eligible to unlock the contents of valuable attributes at a given address (e.g., bitcoin in Bitcoin). The above functions can be completed by a script. When a user provides the output address, he or she can carry a script. Only by unlocking this script by user's private key can the UTXO be unlocked. The blockchain scripting language is a stack-based execution language based on inverted Polish notation. The data structures involved are stacks and provide very limited functionality. In the script, there is no circulation or other complex flow control in case of explosive malicious control flow implantation. The result of the script execution is usually predictable and does not change due to the identity of the executor or other reasons, ensuring the objectivity and correctness of the transaction to a certain degree. The popular scripting languages today are mainly P2PKH, P2PK, MS (15 keys), P2SH, and OP_Return. A transaction can be created by any person with UTXO online or offline, which involves the owner's digital signature to ensure the current ownership of UTXO. To make a transaction effective, it needs to be verified by blockchain network through the common recognition mechanism in the network.

It is to be mentioned that every transaction comes at costs for the user. When a spender transfers money to another account, he needs to pay extra fee. The transaction fee can be used to incentivize the miners to validate the transactions and include them into the blocks they mined. Transaction fees will play the major incentive role in Bitcoin [40]. The fee cannot be too high in case of discouraging the transactions or too low in case of impairing the mining interest. Pass and Shelat [41] argued that cryptocurrencies such as Bitcoin cost relatively high fees, thus making micropayments (payments that are worth several pennies or less than a penny) become more costly. Therefore, they propose a lottery-based mechanism that is able to handle thousands of micropayments per second.

E. Cryptography

There are two main reasons why blockchain needs cryptography. The first reason is to protect privacy. Due to the openness of blockchain, every node in the network can see the complete ledger, which may cause privacy leakage. The second one is to protect the user's assets. Because digital assets cannot easily prove their ownership as physical currency, crypto techniques (e.g., digital signature) are required to prove the ownership of digital assets.

Technically, the cryptographic factors used in blockchain are asymmetric encryption and hash function. Asymmetric cryptography is also called public cryptography, in which a public key is disseminated in public and a private key is kept secret. The hash function can produce a message digest at fixed length for different inputs. In the centralization system, one only needs to interact with bookkeeping center to prove his/her ownership of the property. As long as the center recognizes his/her identity, the proof will be accomplished. However, in a decentralized system, he/she needs to prove his/her identity to the entire network, and ensures the validity of identity through the common recognition of most nodes. It is apparent that centralized identity authentication is no longer applicable. A problem occurs how to prove one's identity without disclosing his/her own password. Asymmetric encryption provides the answer. The user can keep the private key secret but disseminate the corresponding public key to the whole network. The user's identity can be proved by the pair of keys, so as to prove his ownership to the digital property. To solve the privacy issue, pseudonym is needed to achieve anonymity. Specifically, after obtaining the public and private key of identity authentication, the public key can be mapped to the addresses of different users based on mapping rules.

The role of cryptography in the blockchain system has been described above, and the specific processes and details will be briefly introduced in this part. When users need to use their own private keys to sign the transaction while launching a transaction, they have to submit both their own public keys and signatures when submitting a transaction. Each node in the network can authenticate through the public key and signature submitted by the user. Only being verified successfully can the user accomplish the value transfer under his account. The public and private keys of the user are all stored in pairs, and the files that save the key pair are called the user's wallet. The private key is a randomly selected number, while the public key is calculated by the elliptic curve multiplication from the private key. The reverse operation is called "solving the discrete logarithm", which is difficult. One can only do brute search to get the original number. However, it is nearly impossible, thus ensuring the security of the private key.

In a transaction, an account is needed, which can also be called the address of the user. The pseudonym of address provides a good guarantee of the user's privacy. This address can be obtained by a public key through a one-way encrypted hash algorithm. After obtaining this address, it is usually also needed to be coded by Base58Check [42] to make it easy to display. In addition to enhanced readability, Base58Check coding can also compress data and check errors. Take Bitcoin for an example, the process of generating public key to the Bitcoin address is shown in Fig. 5.

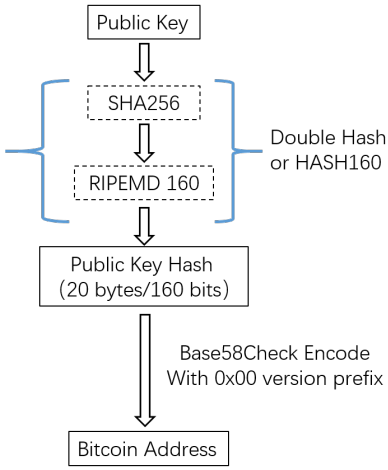


Fig. 5. The process how bitcoin address is generated

F. Blockchain Architecture

Blockchain technology involves many layers rather than simply connecting blocks into a chain. In analogy with the Internet that interconnects information, the blockchain interconnects value. Similar to the Internet protocol suite TCP/IP, we divide blockchain technology into four layers: data layer, network layer, consensus layer, and application layer. The architecture is shown in Fig. 6. In this architecture, each layer plays its own role.

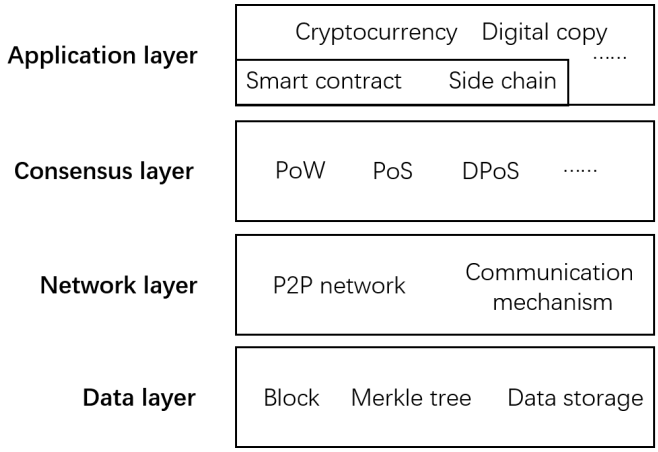


Fig. 6. Blockchain architecture

1) Data layer: Data layer plays the role of designing data structure, organizing data and storing data. Data availability and accessing performance are key factors to be considered. Basic data structure like blockchain is the basic element for storing the key information of blocks and transactions. Different blockchains take different strategies for organizing

data and storing data. For example, blockchain adopts Merkle tree to organize and store the transaction information, while Ethereum takes Merkle Patricia tree. To save disk space, old blocks compaction by stubbing off branches of the Merkle tree is also needed to be taken into account [2]. As for data storage, both Bitcoin and Ethereum exploit levelDB for better data access performance.

2) Network layer: Blockchain is autonomously maintained and managed by a P2P network composed of the miners and users. In a P2P network, there is no center and every node can enter or quit from the network at any time. Every node can also reach any other node with no limitation. In blockchain, the P2P network should tolerate the node failure to a certain degree. All the nodes in this network gain some features, such as equality, autonomy, distributed. They can individually broadcast the mining and transaction information, search the new nodes, etc. Communication and authentication are supposed to be secured in case of attacks.

3) Consensus layer: Consensus is critical in blockchain, since it is managed by a P2P network, wherein each node holds its own view. Making a consensus is a non-trivial task to accomplish, and many consensus algorithms have been proposed to achieve this goal. These algorithms or mechanisms can be classified into two categories: PoW, PoS and their variants; BFT and its variants.

4) Application layer:

- Extension layer. Extension layer aims to extend the ability of blockchain and makes it easier for developers to build blockchain applications. In this layer, smart contract, sidechain, and other techniques are designed to make blockchain more functional. Smart contract, created by Nick Szabo in 1996 [43], is a computer protocol intending to automate the contract execution process in blockchain. Smart contract can eliminate divergence about the agreement terms of the entered parties. Sidechain is a technology that allows digital assets from one blockchain to be used in a separate blockchain and be transferred back to the original blockchain. Though the digital assets can be transferred between the original chains and the side chains, they are isolated. Even one chain crashes, the others will not be affected. Other extensions include Blockchain as a Service (BaaS) [44], fast computing like lightning network [45].
- General Applications. Since being proposed in 2008, blockchain has been applied in various fields, including fintech, insurance, payment, government, etc. Because it is a fundamental technology, it can be combined with many other hot technology, such as artificial intelligence, big data, quantum computing.

In IoT, there is also multi-tiered concept [46], which makes it reasonable to apply blockchain into IoT. Bao et al. [47] proposed a three-tier blockchain-based IoT security architecture. Daza et al. [48] exploited hierarchical and multi-layered blockchains to facilitate finding things or services in IoT. Since the data layer has been described in the above, we will show more details of the remaining layers in the following sections. In later sections, we will also introduce the related IoT concepts.

III. NETWORK

The blockchain exploits an Internet-based P2P network architecture. Each participating node in the network shares a part of computing power, storage capacity, and network connection. These capabilities can become the services that the shared resources provide in the network, and can be accessed directly by peer nodes. There is no need to pass through intermediate entities during the access, thus each node is both a user and a provider of network resources and services. Each network node communicates with each other in a "flat" topology. There are no special status nodes in the entire network. Each node can respond to any peer node and provide resources. For different application requirements, many protocols are also included in the network. For example, the Stratum protocol [49] is an overlay network on top of Bitcoin P2P protocol, aiming to create lightweight Bitcoin clients.

In blockchain P2P networks, the status of each node in the blockchain is equal, and there is no "special node". The successful operation of the whole network is the result of the joint effort of all nodes. The full node in the blockchain generally includes four functions: routing, blockchain transaction storage, mining, and wallet. From the users' point of view, the whole record is not required, only the user related transactions need to be recorded, which also derives the concept of wallet. In most circumstances, the nodes do not need to include all the features and can only include some functions as needed. For example, the user's client can only contain the blockchain part that the user needs, and also does not need to provide mining functions. They provide a Simplified Payment Verification (SPV) [50] way to complete transaction verification. Such a node is called SPV node, also known as "lightweight node". Similarly, some miner nodes are also lightweight nodes, and they must rely on the full nodes maintained by the mining pool servers.

In IoT, there also mounts of devices with data needs to be synchronized via network. In a lot of cases, a device needs to know some nearby nodes' status. Applying blockchain into IoT makes the status and data synchronization process more natural. In [51], Danzi et al. analyzed the communication traffic for blockchain data synchronization of IoT devices. Danzi et al. [52] analyzed the communication traffic for blockchain synchronization of IoT devices. In another work, Danzi et al. [53] stated that to keep a local copy of blockchain ledger on each IoT devices is not feasible. Therefore, they tradeoff the delay and communication for blockchain systems with lightweight IoT devices. Sharma et al. [54] proposed DistBlockNet, a distributed blockchain-based secure Software Defined Network (SDN) architecture for IoT networks. Liang et al. [55] exploited blockchain and traditional cloud server to secure drone communication during data collection and transmission. Kataoka et al. [56] used blockchain and SDN to manage the Internet-wide and distributed IoT traffic. Singh and Kim [57] stated that major issues in Intelligent Vehicle were trust and data accuracy in the communication channel. Therefore, they proposed Trust Bit for Intelligent Vehicle communication using blockchain technology. The Trust Bit is a symbol of trustworthiness of vehicles behavior, and vehicle

legal and illegal activity.

A. Network Characteristics

Some works have done to study the node behavior and other network characteristics. A data collection process is performed in [58], which identified more than 872,000 different Bitcoin nodes. This data allows them to provide information on the size of the bitcoin P2P network, the geographical distribution of the nodes, the network stability in terms of node interruption availability, and some data about the propagation time of the transmitted information. Chen et al. [59] leveraged graph analysis to analyze three main activities on Ethereum: smart contract creation, smart contract invocation, and money transfer. Huang et al. [60] clustered the behavior patterns of all blockchain nodes, and proposed a new BPC (Behavior Pattern Clustering) algorithm. The paper evaluates a series of potential sequence similarity measures and selects distances that are suitable for behavioral pattern clustering problems. Feld et al. [61] suggested a framework traversing Bitcoin's P2P network and generate data about the size and distribution among autonomous systems. Alabi [62] analyzed some blockchain networks to determine whether they meet Metcalfe's Laws. The analysis showed that these networks could be well modeled by Metcalf's law, which defined that the value of a network was proportional to the square of the number of its nodes or end users. A new network model had also been proposed, showing the value to be proportional to the exponential of the root of the network user number, and also showing good consistency The conditions for determining the critical mass based on the new model were also described. Finally, the data from one of the networks was used to discuss and illustrate the potential for identifying value bubbles that can be viewed as a deviation in the value of the model.

B. Cross-chain Network

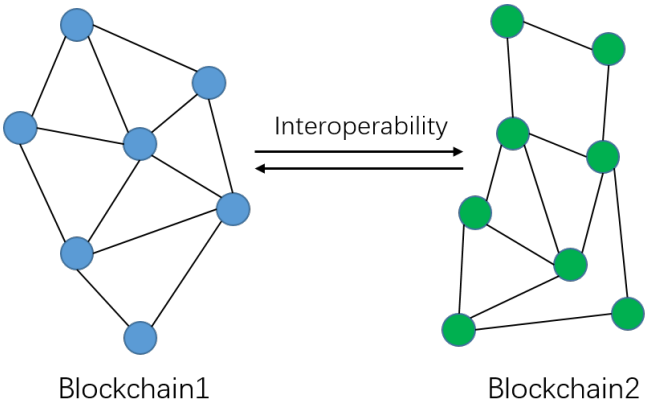


Fig. 7. An example of cross-chain network.

Since there are various blockchain networks, there are needs for these networks to be connected. Fig. 7 shows an example of cross-chain network. In this network, Blockchain1 and Blockchain2 can communicate with each other freely, and the digital assets on both blockchain can be transferred according to the predefined rules.

The early cross-chain technology is mainly based on the sidechain technology represented by Blockstream [63] and the relay technology represented by BTC-Relay (i.e., bitcoin-relay). Its major focus is on asset transfer, while the current project is more concerned with the transfer of chain status. With the development of blockchain, an increasing number of industries have started the exploration of applying blockchain technology in their respective fields. The need for inter-chain operations is urgent, such as collaboration in the supply chain [19] or cooperation between medical services and government agencies or multiple internal links within the company. The Lighting Network [45] is a famous blockchain network supporting cross-chain interoperability. Jin et al. [64] state there are needs to provide "interoperability" between the blockchain systems to enhance their functionality and capabilities. Therefore, they clarify the conception of interoperability according to the cross discipline nature. In their work, they define interoperability as an intelligent characteristic of effective communication and direct information exchange from one blockchain to another, while retains the essence of each individual blockchain, including irreversibility and traceability.There are also other cross-chain networks, which we will introduce below.

Envisioning that the future world would be filled with diverse blockchains serving specific and unique purposes, Polkadot [65] was designed to serve as a network that connects blockchains. Polkadot could enable smart contracts on one blockchain to seamlessly transact with data and assets on other chains. Sun et al. [66] exploited multi-blockchain to build a Central Bank Digital Currency (CBDC) model. In their model, they introduced a cross-blockchain to improve the scalability. It is known as the third generation of unlicensed blockchain technology, and the core concept is solving the major problems that hinder blockchain technology dissemination and being accepted nowadays: scalability and extensibility. Polkadot still focused on Ethereum to realize its interconnection with private chains, and uses other public-chain networks as an upgraded goal, ultimately allowing Ethereum to communicate directly with any chain. The Melonport [67] developed software Melon that will be the first DApp to run on a multi-chain network Polkadot. Melon's ambition was to become a digital asset management platform on a cross-chain framework with more than 700 digital tokens or assets. Its digital asset market value is estimated at approximately US$15 billion, and it is continuing to grow.

In Aion blockchain network [68], the original intention is to provide cross-chain bridges and efficiently manage distributed applications, providing three main functions: integration, expansion, and Spoke. The Interledger protocol creates a system in which two different billing systems can freely transfer money to each other through third-party "connectors" or "verification" machines. The billing system does not need to trust the "connector" because the protocol uses a cryptographic algorithm to create funds for the two billing systems and connectors. When all parties agree on the amount of

funds, they can trade with each other. Meanwhile, only the accounting system participating in the transaction can track the transaction. The details of the transaction can be hidden. The "verifier" runs through an encryption algorithm, so the details of the transaction are not directly seen.

ZhongAn Technology [69] also sees that a single blockchain cannot solve complex problems and bottlenecks in performance relative to traditional centralized systems in many application scenarios. Therefore, it proposes and develops the so-called Anlink, a blockchain network topology. In the Anlink blockchain network, the "parental chain" constitutes the information trunk, and different parent chains exchange information through the link according to the protocol. At the same time, a parent chain carries different sub-chains. These sub-chains can be Ethereum, Bitcoin, and actually can be any kind of distributed ledger implementation. The communication between sub-chains is ensured by cross-chain communication protocols (CBCP). This type of Internet-based layered protocol implementation draws on existing technology implementations to solve the scalability problem of blockchain systems. Early-stage cross-chain technologies such as bi-directional anchoring in Bitcoin and cross-chain smart contracts in Ethernet chains are still used in existing projects. The current cross-chain technology not only inherits the idea of early-stage pledges, but also enriches the new ideas of division of roles, status channels, and trust transfer in the blockchain.

From the analysis of communication means, Polkadot and Anlink use the blockchain itself as the medium for message delivery; unlike the former two, Interledger [70] does not rely on blockchains, but the "connectors" who deliver communication messages to transfer trust; Aeternity uses state channels and status channel routing to support communication between communicators. When using blockchain to deliver messages, one doesn't need to rely on trust to communicate. However, due to the existence of consensus mechanisms, such communication speeds are generally slow; Interledger does not agree on the messages themselves, but is based on neighbors who trust each other to directly exchange messages, which leads to faster delivery speed. Using state channels can reduce the computational pressure on the chain and effectively increase the throughput of the chain.

C. Privacy and Anonymity

Though anonymity is one of the basic properties in blockchain, contributing to the success of Bitcoin, in which users can utilize a number of anonymous addresses for transactions, and many methods have been taken to achieve this goal, it still raises privacy issues concerning the researchers. In many existing blockchains, all transactions are publicly logged, thus raising a number of privacy concerns. Androulaki et al. [31] evaluated user privacy in Bitcoin. Though adopting pseudonyms to protect the user privacy, the public timestamping mechanism of Bitcoin still raises privacy concerns. Actually, because of the public digital signatures, the transaction of individual coins can be tracked. Bonneau [71] claimed they surveyed anonymity issues in Bitcoin and held that the public nature of the blockchain made it possible for attackers to trace the flow of transactions between the pseudonyms and deduce they come from the same user. In IoT, there are also security and privacy challenges. Dorri et al. [72] held that blockchain had the potential to tackle the security and privacy issues in IoT. Golomb et al. [73] utilized blockchain to perform distributed and collaborative anomaly detection for devices with limited sources. A simple example how anonymity is breached can be the multi-input issue in Bitcoin, which is shown in Fig. 8. In this figure, when Alice wants to transfer 4.9 bitcoins to Bob, she may use three public addresses with each one holding some bitcoins (i.e., 1.5 bitcoins in address 1, 2.3 bitcoins in address 2, 1.1 bitcoins in address 3). Then these three addresses can be linked to one identity in Bitcoin. Though Bitcoin takes pseudonyms to achieve anonymity, once any one of the three public addresses is linked to Alice, Alice's anonymity will be breached. This link can happen when Alice puts one of the address publicly to receive donations.

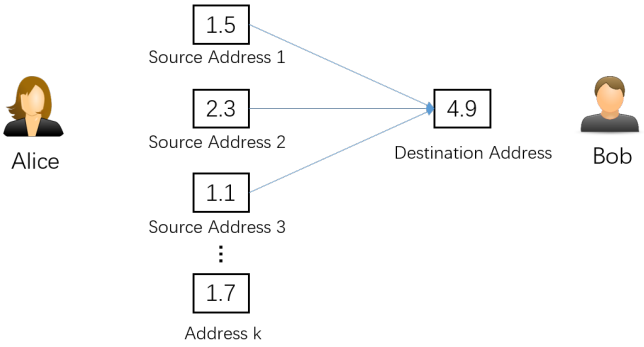


Fig. 8. The multi-input issue in Bitcoin

There are also works attempting to sort blockchain anonymity papers. Herrera [74] grouped bitcoin anonymity papers into three categories: leveraging data to derive information from users including usage patterns; utilizing network information to identify users; proposing mixing techniques to achieve anonymity. Khalilov and Levi [75] classified analysis methods of anonymity and privacy in Bitcoin into four categories: transacting, utilizing off-network information, utilizing network, and analyzing blockchain data. In this subsection, we survey the anonymity papers and put them into two general categories: privacy and anonymity invasion, approaches to achieve anonymity. In each category, we also give our further classification.

1) Privacy and anonymity invasion: We briefly introduce how privacy and anonymity can be released above. We will further discuss the techniques these approaches use in this subsection.

- **Exploiting statistical knowledge**. Ober et al. [32] focused on global network properties and analyzed their implications on the anonymity in blockchain. In their study, they found the merging addresses by simultaneous usage of several addresses brought the greatest challenge to Bitcoin anonymity. They also held that statistical knowledge of dormant bitcoins might reduce the transaction anonymity. Meiklejohn et al. [76] stated the flow of

Bitcoin transaction was globally visible. They developed a clustering heuristic based on changed addresses to cluster addresses belonging to the same user. Ron and Shamir [77] utilized data mining techniques to analyze the public transactions records of Dread Pirate Roberts (DPR) who establish the Silk Road market, and discovered numerous other amounts not seized by FBI. Furthermore, their findings demonstrated the effectiveness of data mining techniques in analyzing publicly available transaction records provided by Bitcoin network. Harlev et al. [78] utilized Supervised Machine Learning to learn the type of entities. They took a sample of 434 entities with approximate 200 million transactions as training samples and classified the unidentified entities into 10 categories, which achieved an accuracy of 77%.

- **Exploiting transaction graph**. Ron and Shamir [79] quantitatively analyzed user activity via Bitcoin transactions graph. In their work, they try to link addresses with the same identity. They held that they can have a full picture of an entity's activities as long as they get the external ownership of any one of merged addresses. One example was that when knowing an address of WikiLeaks who publicized it for donation, they could estimate that Wikileaks owned more than 83 addresses involved in more than 1088 transactions, and it had accumulated 2605.25 BTC of these addresses. Fleder et al. [80] explored the anonymity level of the Bitcoin system. Their transaction-graph-annotation system could annotate the public transaction graph by linking bitcoin public keys to real people. The graph-analysis framework could also trace and cluster user activity. Spagnuolo et al. [81] presented BitLodine, a modular framework, to parse the blockchain and cluster addresses probably belonging to the same user or a group of users. In their work, they even classified these users and labels them for visualizing complex information extracted from Bitcoin network.
- **Exploiting network information**. Reid and Harrigan [82] pointed out the TCP/IP information of the network could be used to reduce the anonymity of blockchain. One case was that when one person pays change to himself, multiple public keys of himself would appear in the same transaction, which could be exploited to link a user. Koshy et al. [83] also developed heuristics to identify the ownership between Bitcoin address and the corresponding IP addresses. In their study, they found 1000 bitcoin addresses can be mapped to their possible IP addresses by leveraging anomalous relaying behavior. Alex et al. [84] studied the IP traffic of the Bitcoin peers would reveal the origins of transactions and disclose the identities of users. Their work showed that even when users took NAT (Network Address Translation) or firewall, they could still deanonymize them and link the pseudonyms and IP addresses. Steven et al. [85] exploited the web cookie that kept the shopping records to link a user's real identity with the transaction addresses in cryptocurrency systems even when this user took anonymity techniques, thus laying great risks to people's privacy. Observing that Bitcoin was an anonymous system and many criminal activities are committed there, Moser et al. [86] utilized reverse-engineering techniques to trace the anonymous transactions related to their probe accounts.

2) Approaches to achieving anonymity: Based on the analysis of the above problems, a lot of researchers have put forward numerous methods to improve anonymity in blockchain systems. These methods can be divided into three categories: (1) to introduce intermediate trusted institutions to complete the address change, reducing the exposure rate of user privacy; (2) to propose some improvement measures the existing system to enhance the system security; (3) to invent new methods by cryptographic techniques.

- **Laundry service with intermediary**. The first method is to exchange money with laundry service to reduce the connection of information in its account, and now there are still agencies that use it [87], [88]. Obviously, this approach has a more serious limitation: operators may steal money, track money, or have gone bankrupt. Heilman et al. [89] also presented solutions to the anonymity problem by using an untrusted third party to issue the anonymous voucher, and ensure the anonymity and fairness during bitcoin and voucher exchange. Heilman et al. [90] presented a new unidirectional unlinkable payment hub named TumbleBit to achieve fast anonymous off-chain payments. It introduced an untrusted intermediary called Tumbler. Camenisch et al. [91] found that in the certification of keys and attributes, the privacy of issuers could be released. Delegatable anonymous credentials could address this privacy leakage problem. Therefore, they claimed to propose the first practical delegatable anonymous credential system which is implemented for the transaction authentication of blockchain. The criminality possibility caused by decentralization and anonymity of Bitcoin is also discussed in [92]. Through implementing anonymity mechanisms, the authors still can link the input and output transactions in BitLaundry [93], a Bitcoin mixing service to unlink the bitcoin transactions and addresses. Bonneau et al. [94] proposed a mixed protocol called Mixcoin to achieve strong anonymity in Bitcoin. Green et al. [95] propose three channel payments to achieve privacy, anonymity, scalability and storage issues via untrusted intermediaries.
- **Improvements based on existing systems**. Ruffing et al. [96] found that most of existing P2P anonymous communication protocols suffer from slot collision (i.e., two peers send messages at the same slot) and malicious peers, and the remaining needs $O(f^2)$ communication rounds when f malicious peers exist. Therefore, they proposed DiceMix, which is a cheaper anonymous communication protocol with $4+2f$ rounds at worst. Bissias et al. [97] stated that the movement of coin between addresses can be observed by examining the public blockchain, which was a fundamental limitation of Bitcoin and its variants. Considering this, they proposed Xim [97], a decentralized protocol to achieve anonymity

while thwarting Sybil attacks, DoS attack and timing-based interference attack various attacks. Cecchetti et al. [98] presented Solidus, a protocol that can hide the transaction values and identities of entities while keeping the transaction publicly verifiability. Miers et al. [99] took zerocoin to extend bitcoin, and proposed a decentralized e-cash mechanism. Also, they designed the zerocoin as an instance and checked its security, in which they took cryptography technology without introducing third party trust mechanism, cut off the transaction between the contract, and enhanced the security of the system. Based on zero-knowledge Succinct Non-interactive Arguments of Knowledge (zk-SNARKs), Sasson et al. [100] proposed Zerocash, a mechanism could hide destination source, payment source, and transfer number in the transaction. They also compared Zerocash with bitcoin and stated that the information hiding could not encourage tax evasion. Therefore. The security was also enhanced.

- **Exploiting cryptographic techniques**. The third promotion method exploits the cryptography techniques to keep the anonymity and security. Cryptonote is a mechanism different from bitcoin, using sing signature technology as the encryption mechanism of blockchain. Ring signature was proposed by Rivest, Shamir, and Tauman. It is a simplified group signature. In ring signature, there are only ring members but the manager, and no cooperation between members of the ring is required [101]. The signer first selects a temporary signer set, with the signers included. The signer can then use his own private key and the public key of other people in the signature set to generate the signature independently without the need of others' help [102]. In [103], T. Okamoto and K. Ohta showed their ideas about the privacy in ideal e-currency. Based on this idea, [104] extracted two properties: untraceability and unlinkability. The untraceability requires that all possible senders are equiprobable for each incoming transaction. In another word, one cannot determine who is the sender when catching a sending message. Unlinkability requires it is impossible to prove they are sent to the same person for any two outgoing transactions. It is clear that bitcoin does not meet the requirements of untraceability.
  Monero is an application based on Cryptonote. In [105], Kumar et al. also gave the similar definition of unlinkability and untraceability in Monero. In order to achieve unlinkability, Monero brings the notion of *one-time random addresses*, as is shown in Fig. 9. The untraceability is ensured by using ring signatures. In the figure, the created Ring signature can hide the real input key corresponding the output being spent. A series of articles about Monero can be found on the website of Monero Research Lab. Noether et al. [106] analyze the scenarios of Cryptonote dealing with plausible attacks. On Sep 4th, 2014, there was an attack against the Monero encryption system network. Macheta et al. [107] discussed this specific situation of the attack, and introduce some initial compensation measures for Cryptonote community and Monero community. Noether et al. [108] introduced the encryption process involved in Monero in detail from the mathematical principle to the application implementation. Msackenzie et al. [109] pointed out several attacks that can pose a threat to the security of Cryptonote protocol and proposed improvement measures to enhance their stability. Noether et al. [110] proposed the improvement measures for the existing encryption mechanism of Monero, and the amount in the transaction process can be hidden by using the signature proposed in [111]. Mrlring [112] gave a modification of gmaxwell's Confidential Transactions to ring signatures. This improvement can be applied to two Monero models in [108] and [110]. Kumar et al. [105] conducted a forensic analysis of the Monero blockchain. After implementing and evaluating the attacks on the Monero blockchain, they demonstrate the security of the blockchain system.

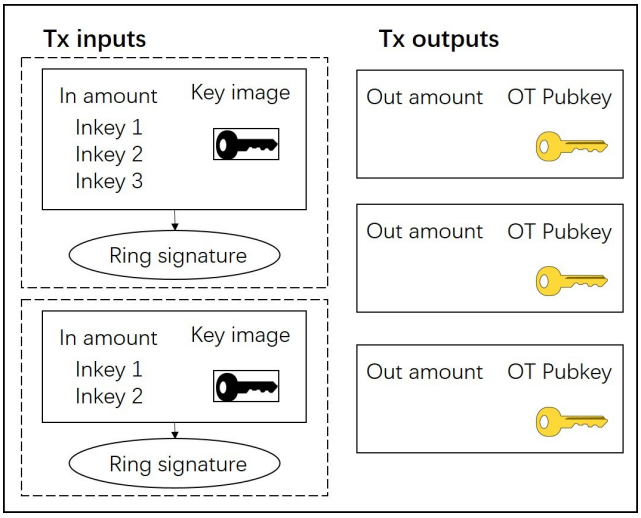


Fig. 9. A representation of a simplified Monero transaction. It has three two inputs and three outputs. The first input uses two mixed public input keys (hence three input keys) and the second uses one. The sum of all input amounts must equal the sum of all output amounts. The ring signatures can hide the real input key according to the output being spent. The key image is used to prevent double-spending attack. [105]

In addition to the above discussions, anonymity may facilitate the illicit activities because of its untraceability. Bancroft and Reid [113] conceived of anonymity as a way of engaging and maintaining social relationship in an anonymous mode. They carried out a study of darknet 'cryptomarket' users who mainly leveraged the darknet to trade illicit drugs, discuss drug quality and share information on safe and effective use. They also held that attempts to promote deanonymizing norms and technology were based on an erroneous understanding of anonymity. Matzutt et al. [114] observed that every peer in Bitcoin needed to locally store any transaction data in a persistent way, thus keeping the non-repudiation property of blockchain. However, this property could be abused to propagate arbitrary or even harmful data.

D. Attacks and countermeasures

There are three major threats to the blockchain network: Sybil attack, Eclipse attack, and DDoS attack. Also, researchers further propose other novel attacks. Xu et al. [115] gave a basic investigation of attacks against blockchain and conclude that further research is needed to improve the security of blockchain.

1) Sybil Attack: The Sybil attack was first proposed by Douceur in P2P networks [6]. In Douceur's work, he stated that Sybil attack can destroy the redundancy mechanism in distributed storage systems. To thwart this attack, he even proposed authentication suggestions in the absence of an identity authority. Later, Karlof and Wagner [116] pointed out that the Sybil attack also posed a threat to the routing mechanism in the sensor network. In Sybil attack, a malicious node illegally presents itself with multiple identities called Sybil nodes. A Sybil attacker can fill the blockchain network with the fake nodes he control, resulting in some victim nodes being isolated. Sybil attacks mainly include the following types: direct communication, indirect communication, forged identities, stolen identities, simultaneous attacks, and non-simultaneous attacks. In a blockchain network, users do not need to pay to create new identities or new nodes. Attackers can use this loophole to launch Sybil attacks, forge their own identity and join the network. After obtaining the identity of several nodes, attackers can mount this attack randomly to break the blockchain network. For example, misleading the node routing table, reducing the search efficiency of the blockchain network node; or transmitting non-authorized files in the network, destroying the file sharing security in the network, consuming connection resources between nodes. Otte et al. designed TrustChain [7] to offer scalability, openness, and Sybil-resistance while replacing PoW with a mechanism to establish the validity and integrity of transactions. The Sybil-resistant algorithm in Trustchain is named NetFlow, whose role is to determine the trustworthiness of agents in an online community.

2) Eclipse Attack: Singh et al. proposed the Eclipse attack [117] on overlay networks. In blockchain networks, Eclipse attackers can insulate victim nodes from the normal blockchain network by stealing the routing table of the node and adding enough fake nodes to the set of neighbor nodes. When the victim node is attacked by Eclipse, most of its external route paths are controlled by the attacker nodes. What is worse, the attacker nodes can further take vicious actions, such as route fraud, storage pollution, denial of service, and ID hijacking. Therefore, Eclipse attacks pose a serious threat to blockchain networks. The normal operation of the blockchain network relies on the sharing of routing information between blockchain nodes. The Eclipse attacker influences the routing table of blockchain nodes by sending routing table update messages to blockchain nodes constantly, trying to make the routing table of ordinary nodes full of spurious nodes. When the falsified nodes in the routing table of the blockchain node occupy a high proportion, its normal behavior of the blockchain network, including route lookup or resource search, will be isolated by attacker nodes, and this is the reason why the attack is called the Eclipse attack.

The Eclipse attack is closely related to the Sybil attack in which multiple malicious nodes or identities are forged. It usually needs more Sybil nodes to mount an attack. In order to achieve Eclipse attacks on a specific blockchain node group, an attacker must first set up enough Sybil attack nodes and declare them to the blockchain network as "normal" nodes, then exploit these Sybil nodes to communicate with normal nodes and invade their routing tables, thus eventually isolating them from the blockchain network.

Two major works have been done to study the Eclipse attacks on blockchain networks. Heilman et al. [118] specifically introduced the Eclipse attack in bitcoin's peer-to-peer networks. For blockchain networks, the Eclipse attack destroys the topology of the network, reduces the number of nodes, and greatly reduces the efficiency of resource sharing. In extreme case, it can completely control the entire blockchain network, dividing the whole network into several independent blockchain network areas. For the victim blockchain nodes, they are out of the blockchain network without awareness. The consequences are that all blockchain network requests are hijacked by attackers, and most of the replies they receive are falsified and normal resource sharing or download cannot be performed. Nayak et al. [119] combined stubborn mining attack with eclipse attack, and gain more 30% gains compared with the naive eclipse attack.

3) DDoS Attack: DDoS (Distributed Denial of Service) attack is one of the most threatening attacks for blockchain network. In DDoS attack, the attacker exploits the Client/Server (C/S) model to combine multiple computers as an attack platform to launch attacks against one or more targets, thus multiplying the power of denial of service attacks. Since there are millions of concurrent online nodes that holding a large quantity of resources (e.g., the storage, bandwidth) and a blockchain node needs to keep a copy of the whole network, blockchain network can be used as a DDoS attack engine. According to the different attack strategies, blockchain-based DDoS attacks can be divided into active attacks and passive attacks.

In active DDoS attack, the attacker actively sends a large amount of false information to the network node, so that subsequent visits to this information will be forwarded to the victim, thus achieving the effect of DDoS attack. This type of attack utilizes a push-based mechanism in the blockchain network protocol. Reflective nodes receive a large amount of notification information in a short period of time, which is not easy to be recorded and analyzed. Also, it can avoid IP checks by using fake source addresses, making it more difficult to track and locate attack sources. In addition, active attacks that introduce extra traffic in the blockchain network will reduce the search and routing performance of the blockchain network. Blockchain-based passive DDoS attacks passively wait for queries from other nodes by modifying the blockchain client or server software, and then return false responses to achieve an attack effect. Normally, some amplifying measures will be taken to enhance the effectiveness of the attack, such as deploying multiple attack nodes, including multiple target hosts in one response message, combining other protocols, or

implementing vulnerabilities. This attack exploits the "pull" mechanism in the blockchain network protocol. Passive attacks are non-intrusive and have little impact on the blockchain network traffic. Usually, they can only be used as local blockchain nodes.

Sybil attack and Eclipse attack can facilitate DDoS attacks in blockchain networks. The goal of the Sybil attack is that a single physical node generates a large number of different identities on the blockchain network. Successful Sybil attacks can make it easier to launch Eclipse attacks. The precondition for DDoS attacks on a single node is to issue a large number of false messages to the blockchain network or passively make false responses. Eclipse attacks can help attackers hijack the information passed between network nodes and increase the likelihood of successful DDoS attacks. The Sybil attack is just posing as a single blockchain network node, and the impact on the blockchain network is relatively small; the Eclipse attack makes part of the blockchain nodes away from the blockchain network, which is unacceptable to the attacked node. The purpose of the DDoS attack is to occupy a large number of resources of the victim node and prevent it from providing services normally. Therefore, the impact of the DDoS attack on the blockchain network is fatal.

To thwart DDoS attacks, there are mounts of works. Back [8] suggested *hashcash* and claimed it could throttle DoS against remailer networks and deter email spam, as well as other applications. In *hashcash*, the CPU cost-function could compute a token that can be used as a proof of work. Plohmann and Gerhards [120] provided an overview of the "Miner Botnet" that was botnet being controlled to mine cryptocurrency. It also could be used as a flexible toolset for various illegal activities including DDoS attack. In their work, they even introduced the DDoS-related modules like *ddhttp.exe*, *edp.exe* and *pele.exe*. Vasek et al. [121] stated that they documented 142 unique DDoS attacks on 40 Bitcoin services between May 2011 and October 2013, and 7% of all known operators had been attacked. In their study, they found that big mining pools with high hash power were much more likely to be DDoSed than small ones, which was also found by Johnson et al. [122]. In addition to this finding, Johnson et al. [122] also observed that larger mining pools had a greater incentive to attack than smaller ones. In their work, they utilize game-theoretic methods to analyze DDoS attacks against Bitcoin mining pools. Laszka et al. [123] also noticed that attackers might diminish the mining power of competing pools through DDoS attacks. They held that the traditional works [121], [122] were limited in analyzing the immediate impacts on the mining pools rather than long-term effects, which had a strong influence on the behavior of service providers. Therefore, they developed a game-theoretic model for investigating the long-term impact of attacks.

Though being attacked by DDoS attack, blockchain technology can also be used to mitigate or defense DDoS attack. Defending DDoS attack is not a easy job that can be accomplished by individual systems for its large-scale and brute force, especially for small ones with limited resources. However, the existing defense mechanisms are limited in making a coordinated and distributed defense fully operational [124], [11]. In [124], Wang et al. deploy anti-honeypots in servers to detect and prevent DDoS attacks by a two-step anti-honeypot-based strategy. In [11], Rodrigues et al. designed an architecture that combines blockchain and smart contracts for flexible and efficient DDoS mitigation solutions across multiple domains. The architecture can fully exploit the existing public and distributed infrastructures to defend DDoS attacks.

4) Replay Attack: In the traditional computer, replay attack is a kind of attack in which a valid transmission is maliciously repeated. In blockchain technology, replay attack occurs when the blockchain is hard-forked and a transaction on one chain is replayed on another since both transactions are valid. Because of the hard bifurcation of the two chains, their addresses are the same as the algorithm used to generate the private key, and the transaction information is exactly the same, thus resulting in transactions on one of the chains that are likely to be perfectly legal on the other. If one initiates a transaction on one of the chains and broadcast it, the attacker can go to the other chain and replay it by utilizing the transaction details. For example, Bitcoin has forked its bifurcation, leaving rooms to replay attacks, which has been widely discussed. The original bitcoin chain is referred as BTC, and the other chain that bifurcates as BCC (Bitcoin Cash). Then we know that BCC completely inherits BTC information before bifurcation. Therefore, when a transaction processes on the BTC chain, another transaction with the same details can also occur on the BCC chain. It means after spending 1 coin on the BTC, then spending once again on the BCC. To thwart replay attack in blockchain, introducing a nonce in each request is a common design [125], [126], [127].

IV. Consensus

A consensus mechanism in blockchain is a mechanism to achieve agreement about a value or network state among peer nodes in the network. It is a fundamental problem that needs to be solved in distributed systems. Blockchain consensus mechanism can be used in IoT. The disadvantages in IoT applications, such as high maintenance costs, weakness for supporting time-critical usages, can be solved by introducing a proper distributed consensus mechanism. Sagirlar et al. [128] applied PoW in IoT system design to overcome the above issues in blockchain. However, Salimitari and Chatterjee [129] stated that though PoW appears not practical for IoT networks because of its high computational and bandwidth requirements. In their work, they also discuss the possibility of applying other consensus mechanisms in IoT network, such as PoS, DPoS, PBFT, which we will introduce later.

Gramoli [130] discussed the mainstream blockchain algorithms and compare different consensus problems tackled by blockchains. Wang et al. [131] claimed there lacks comprehensive literature review on the blockchain development, and their work focused on consensus system design and incentive mechanism design.

A. The Distribution Essence

1) Trust and consistency: The main reason for the prevalence of blockchain is that it solves the problem of trust. The

trust that appears to be easily accessible is not actually easy. Trust in reality often comes from the long accumulation of the government agencies, authority institutions, or good reputation. When we have a transaction or property transfer, a third party is often needed to do notarization and high costs are also paid for proving. The transaction is conducted between two parties, and the details of the transaction should not be exposed to the third one. Also, high costs are not supposed to be paid to finish the transaction. Under these circumstances, it is natural to lead the centralization of the authoritative party system to a multi-node distributed system without trust. The demand of the distributed system is to weaken the role of the centralization authority, which is slightly different from the traditional design of the distributed system based on computing power or other resources. Therefore, more attention is paid to whether the process of centralization is secure or credible in blockchain.

2) Ideal consistency: In a traditional distributed system, one of the most important topics is to consider the consistency problem. Through negotiation and cooperation, many nodes present a unified state to the outside world, and the system can provide stable services to the outside world. Achievements have been made in the past decades of research on consistency. Consistency means that for a number of service nodes in the system and a series of given operations, making them achieve the same result under the protection of protocol. Ideal consistency usually meets three conditions: terminability, consensus, and legality. Terminability refers to the consistent result in a limited time. Consensus means that the final decision of different nodes should be the same. Legality means that the result of the decision must be a proposal put forward by other processes. It is obvious that simultaneously achieving these three properties is at high costs. That is the reason why it is called ideal consistency. Strong consistency can also be divided into two categories: sequence consistency [132] and linear consistency [133].

3) Application consistency: In an actual application environment, there is often no need to achieve such a high requirement of consistency. Therefore, some of the property can be put down. For most distributed systems, there is always a time when we can achieve final consistency under certain constraints. In real distributed systems, there exist three problems: 1) the communication of the nodes are not reliable, including random delay and content fault. 2) node processing is possible to be wrong or even crash. 3) synchronous call makes the system lack extensibility. The solution is usually to serialize the parallel operations that may cause inconsistencies. However, after Fischer's demonstration, there is no deterministic algorithm that can solve the consistency problem in a reliable network with node failure (even if there is only one) [134]. Though this theory has proved the impossibility of strong consistency, in engineering's perspective, we can get the desired result after weakening different characteristics according to different scenarios.

4) Traditional consensus algorithm: In Gilbert and Lynch's work [135], they stated that it is impossible to achieve consistency, availability, and partition at the same time in a distributed system. When designing a system, one of the three properties needs to be weakened. Specifically, when it comes to algorithm design, problems in the two scenarios are supposed to be solved. One is the failure of nodes (not responding), which is called "non Byzantine error". The other is the situations of malicious response in nodes, called "Byzantine error" (the corresponding node is Byzantine node). To deal with Non Byzantine errors, Paxos [136], Raft [137] and their varieties were proposed. For Byzantine errors, one representative can be PBFT (Practical Byzantine Fault Tolerance) [138]. Although the blockchain system is also a distributed system, they are slightly different from traditional distributed systems. First, traditional distributed consensus algorithms usually do not consider Byzantine fault tolerance, which assumes that all nodes will only go through downtime, network fault, and other non-human problems, while other malicious nodes that tamper data are not included. Second, the traditional distributed system is mainly oriented to log (database), while the blockchain model is transaction oriented. Therefore, the traditional distributed consensus algorithm can be regarded as the next layer of the consensus ones in blockchain. Compared with traditional distributed systems, blockchain involves more challenges to be handled. For example, in blockchain, it needs to solve the unique consistency problem for trading—double-spending [2]. In the design of blockchain system, double-spending is the first issue to be resolved. Blockchain system also needs to deal with the attacks mentioned above, such as Sybil attack, Eclipse attack, DDoS. From the PoW algorithm originally born in bitcoin to the classical improved algorithm of BFT, the initial proof is weakened. Therefore, the content related to the consistency of blockchain and other distributed systems is collectively referred to as a consensus mechanism.

B. PoW

Since bitcoin has been released for eight years, with the rapid development of bitcoin, the blockchain technology behind it has become increasingly important. In the initial blockchain design, the common understanding mechanism used by the bitcoin system is PoW, which comprehensively considers the above problems and challenges. Simply speaking, every node in the system provides the computing power for the whole system, and through a competition mechanism, the best node can solve the computation puzzle and get the reward of the system. That is to say, the distribution of the newly generated money is completed.

In the above, we have mentioned Nakamoto uses asymmetric cryptography to solve the problem of ownership of cryptocurrency, uses block transaction time stamp to solve the transaction existing problem, and uses distributed ledger to solve the verification problem for third party structure after the transaction. Then the only problem remains is the correctness and uniqueness. The correctness and uniqueness need to ensure that there is no double payment and no malicious node to modify the ledger. After the decentralization, the task of keep ledgers falls on each node. It is the significance of the consensus mechanism to ensure that each node can complete the work of keeping ledgers in accordance with the established rules. Therefore, in order to add cost and reward to the bookkeeping process, Nakamoto comes up with the

idea that one should pay a certain price to solve the hash puzzle to get the right of bookkeeping. After paying that price and keeping ledgers according to the rules, the miners can get corresponding rewards. To put it simply, PoW is forcing every node to insert a number of nonce into a block before making every submission to the next block, making $F(nonce) < Target$ where F is a hash function, and $Target$ is a parameter determined by a difficulty function. Determined by the properties of hash function, we must achieve the goal of finding the qualified nonce by an exhaustive nonce method, thus completing the goal of providing workload. In addition, the successful operation of PoW requires the assistance of two other rules:

- The longest chain rule: take the longest chain as the right chain.
- The motivation rule: one will be rewarded by finding a proper block.

With the two principles, the PoW enables the bitcoin system to reach a Nash equilibrium, so it is able to run smoothly. This most essential basic protocol, PoW, has been deeply dissected from many angles and many aspects, and the following will be introduced one by one.

Works for PoW analysis can be roughly divided into several parts: modeling the consensus algorithm, analyzing of the characteristics of the consensus algorithm, and improving algorithm security against attacks.

1) Consensus Process: In Nakamoto's model, the PoW's consensus algorithm is introduced in a vague and qualitative way, and it is not accurate and convincing in the follow-up analysis. For this reason, some researchers are modeling the PoW consensus algorithm, hoping to make the system architecture more clear and convenient for subsequent analysis through formal description [139][140][141][142][143]. The modeling of this blockchain system can be divided into two types: network modeling, blockchain ledger modeling. The former is usually modeled as a directed graph [144], and the vertex is a node in the network, while the edge represents the communication delay between the two nodes. The latter can usually be divided into chain ledger or tree-structured ledger [145] [141] [146]. In the refined model, related concepts can be divided into two parts. The first part contains the network parameters including miner's computation power ratio q (honest for q, not honest for $1-q$), time t (the definition of time t can be varied, such as the current mining time, usually according to the modeling of the need), the speed of mining s (approximately take the production of blocks as a single Poisson process), the communication ability of miners γ. The other part contains the ledger parameters including the number of blocks z needing to be waited to confirm the transaction, the value v of the transaction, the depth of the block d. Kiayias et al. [147] built their model in perspective of the miners, with the mining processes simplified into two forms of game. The first game was relatively simple, in which miners were not sure which block to mine, and they should release it once mining successfully. The second game was a little more complex, in which the miners chose which blocks to mine, but when the miners released the blocks is uncertain. These two games illustrated when each miner's computing power was relatively small, their best response matched the expected behavior as the PoW designer. However, when the miners had great computing power, they deviated from the expected behavior and had other Nash equilibrium. Kroll[148] et al. simulated bitcoin mining process as games for miners and bitcoin holders. They considered the "51%" effect in the angle of game theory, and put forward the necessity and importance of bitcoin management. Lewenberg et al. [149] analyzed game theory about the cooperative mining of the mining pools.

2) Analysis of the characteristics of consensus algorithms: Gervais et al. [140] introduced a quantitative framework to analyze the security and performance of different consensus and network parameters in PoW based blockchain. This framework allows for capturing existing PoW based deployment and PoW blockchain variants instantiated with different parameters, and objectively comparing the trade-off between their performance and security characteristics. Pass et al. [142] defined the abstract representation of a blockchain protocol and determine the security features in these protocols. The most important thing is to increase the analysis of asynchronous networks, which shows that the PoW consensus mechanism meets strong consistency. Vukolic [150] made a comparison of the scalability and performance between PoW and BFT variants. He found PoW gains advantages in scalability (i.e., number of nodes, number of clients) but disadvantages in performance (i.e., latency, throughput). Chase et al. [151] proposed an abstract of transparency overlay, which they claim could be used with any system and gave provable transparency guarantees. It could eliminate the mining need and entire blockchain data storage for users in Bitcoin.

C. PoS

PoW is now the most common consensus algorithm used in many blockchain applications. Obviously, its success attributes to bitcoin in the digital currency. Although bitcoin system has obvious advantages, with the increase of application and research, the defects of PoW consensus algorithm used in bitcoin system are more and more obvious. There are many drawbacks in accumulating trust through a lot of computing power. In addition to the security problems caused by attacks mentioned in the above section, there are also problems including too low throughput, too much waste of resources. Therefore, researchers are actively studying the alternative consensus algorithms, which not only guarantee the safety of distributed ledgers, but also reduce the consumed resources and improve the throughput of transactions. At present, there is a more mature algorithm named PoS (Proof of Stake). Simply speaking, this mechanism calculates the percentage of the total number of money that the competition node holds, and it can also include the time that the node takes the number of coins to decide the probability of acquiring the right of account. In order to enable each block to be generated faster, the PoS mechanism eliminates the exhaustive nonce process, and then adopts the following faster algorithm: $H(H(B_prev), A, t) \leq balance(A)m$, $F(Timestamp) < Target \cdot Babance$, where H is still a hash function, t is UTC timestamp, B_prev refers to the last block, $balance(A)$ represents the account balance of

account A, and m still represents a certain defined number. On the left, the parameter can be adjusted by T, and the right m is a fixed real number. Therefore, the greater the $balance(A)$ is, the greater the probability of finding a reasonable t is. Note that the parameter in the left of inequity changes from nonce to timestamp. Obviously, the search space is significantly reduced, which can improve the efficiency of mining.

1) PoS algorithm modeling and analysis: The consensus algorithm is mainly analyzed from the theoretical point of view in Snow White [3] and Ouroboros [152], [153]. The process of finding consensus can be seen as the process of electing a leader, based on the goals that PoW has to achieve. In Ouroboros [152], the PoS algorithm is abstracted formally, and the concept of slot time is introduced to link each block. For each node, the current time can be synchronized by one clock. In order to meet the persistence and activity of blockchain, the whole protocol can be divided into three parts: Common Prefix (CP), Chain Quality (CQ), Chain Growth (CG). The state of the system is divided into four categories according to the different restrictions on the dynamic and static property rights, and whether there is an authorization or not. After the whole system executes an operation in accordance with the protocol, its state can be represented as $VIEW^{P,F}_{\prod,A,Z}(k)$ in the view of party P, where $\prod$ is the protocol, A is the adversary, Z is environment, k is security parameter and F is ideal functionality. In addition, the details of the whole system are formally defined, such as block, blockchain, period, feature string, etc. The security of the three aspects of the system (i.e., CP, CQ, CG) can be verified by these formalized definitions. The authors in Snow White [3] take an alternative way to prove the security of PoS. They do not directly model the PoS protocol itself. Instead, they propose a reconfigurable consensus and put forward requirements different from the traditional consensus mechanisms. By virtue of this idea, the first instance of the reconfigurable negotiation protocol is created, and the example is used to show how to implement PoS. The security of the entire system is also given in the appendix section from CP, CQ, and CG. However, Meiklejohn et al. [76] and Barber et al. [154] described the situation that stakeholders did not believe in monetary value, which would threaten the security of the system.

2) The application of PoS: The concept of PoS first appeared in the bitcoin community post Quantum [155]. This concept is a good solution to the problem of mining tragedy proposed by Vandroiy in the forum [156]. Although the idea had been proposed at the time, bitcoin did not immediately use this consensus algorithm. It is not until 2012 that PoS consensus algorithm is first implemented in the Peercoin. Peercoin proposes a new concept called coin age, and its size is determined by the product of cash number and cash holding time. The mining formula is $profhash < coinage \times target$. Once obtaining the right to create a block, the previous age will be cleared and recalculated.

In Peercoin, PoW is used for early mining and distribution of cryptocurrency, and PoS is applied later. Therefore, blocks are also divided into two types: PoW based block and PoS based block. The process of stake proof can be found from the content of PoS based blocks. There will be a special transaction in PoS based blocks similar to the coinbase in PoW based blocks, which is called coinstake, as is shown in Fig. 10. The kernal input contains factors such as the age, the target value, the random number to meet the demand of hash function, which is similar to the mining process of PoW. Because of its limited search space, the difficulty of solving hash puzzles in PoW is much less than that. In addition to the mining process, the coinage process based on PoS is also different. It produces interests based on the age of coin consumed in coinstake, and the annual interest rate set in the Peercoin is 1%.

Although the concept of mining using currency age reduces the new block production difficulty, it also has some problems. For example, the resistance against double-spending attack is not strong. In PoW, the cost of generating bifurcation is large, while the cost in PoS is weakened. The nodes create multiple bifurcations with almost no loss, which hinders the system consensus. A checkpoint broadcast is used in Peercoin, and the height of the blockchain is not allowed to be higher than the known final checkpoint.

After the application of the PoS mechanism in the currency, there are a lot of cryptocurrencies using PoS consensus, such as NVC [157], YAC [158], CMC [159]. NVC utilizes script instead of the hash algorithm to make the transaction faster, raising the annual interest rate to 5%. YAC has made a major breakthrough on the basis of NVC, and it adopts a new script-Jane algorithm. One of the characteristics of the algorithm is that CPU's dependence is stronger than that of GPU. It has been supported by a large number of users. Based on NVC, CMC adds a transaction message function and also uses the script algorithm. Compared with NVC, CMC has an annual interest rate 1.5%. Different from the above currencies, Nxt [160] is not based on bitcoin development, and is the first digital currency using pure PoS consensus algorithm, but its consensus algorithm is different from that in Peercoin, which is $hit < baseTarget \times effectiveBalance \times elapseTime$. Nxt also provides the function of "transparent forging", enabling users to automatically decide which server nodes can generate the next block. It ensures the system's security by punishing the inactive nodes. According to the algorithm, after the emergence of a new block in the network, each user's hit value is fixed for forging next block, and the target value is proportional to the effective balance of accounts. BLK (Blackcoin) [161] puts forward the concept of PoS 2.0. It removes coin age used in the Peercoin, and uses the following algorithm to select miners: $proofhash < coins \cdot target$. This approach can make nodes keep online as much as possible, while nodes in Peercoin can accumulate coin age even when they are offline, which provides opportunities for bad nodes. In addition, in order to prevent pre-calculated attacks, the stake correction factor is changed every time. The last difference is that the hash algorithm still uses SHA256 in BLK. Table I compares the three main currencies.

D. PoW and PoS Variants

PoW and PoS are the basic consensus algorithms in blockchain. Both of them have been widely used in many

TABLE I. A COMPARISON OF PEERCOIN, NXT, AND BLACKCOIN

	PPcoin	NXT	Blackcoin
POS	proofhash < coin age × targetValue	hit < baseTarget × effectiveBalance × elapseTime	proofhash
is pure POS ?	no	yes	no
crypto algorithm	SHA256	SHA256	SHA256
need mining?	yes, there stages: PoW, PoW and PoS, PoS	no, generate one billion NxT coin in the first block	yes, there stages: PoW, PoW and PoS, PoS

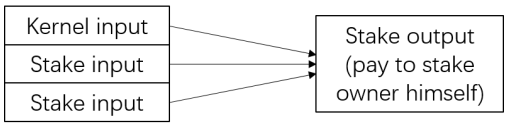


Fig. 10. Structure of Proof-of-Stake (Coinstake) Transaction

applications. Most of the other consensus algorithms are proposed based on these two algorithms. In this section, we will give a sight of the derived consensus algorithms in blockchain.

- DPoS (Delegated Proof of Stake)

DPoS algorithm is similar to the modern enterprise board system. It solves the centralization problem by introducing the delegate mechanism [4]. Each block is generated by the delegate who signs his or her signature, and the delegates are voted by the nodes of the blockchain network. Through selecting the delegates, the DPoS algorithm eliminates the need for the transaction to wait for the confirmation of the untrusted nodes. By reducing the confirmation time, the speed of the transaction has been greatly improved. DPoS algorithm can be understood as a combination of centralization and decentralization. Through the election, everyone is possible to become a delegate representing the vast majority of users. If some delegates violate the protocol, their rights to write the ledger will be denied, and the network can reelect new nodes to replace them. DPoS can guarantee correctness under various circumstances, such as a few bifurcations, a few off-line multiple productions, network fragmentation and other related issues [162], [163]. At present, the systems using DPoS include BitShares, Steem and EoS. The delegates number of these three applications is 101, 21, and 21 respectively. DPoS gains some advantages over PoW and PoS. Compared with PoW, DPoS is faster and more efficient. Also, it is more democratic and flexible than PoS. Delegates can be selected fairly and bad delegates can be kicked out quickly. The disadvantages are that it is not as decentralized as expected for the existing the trustworthy delegates. Also, people are reluctant to vote unless there are enough incentivized. In some cases when there are not enough votes, large stakeholders can easily dominate the voting to gains benefits.

- PoL (Proof of Luck)

According to the design mechanism of PoW, each node needs to pay huge computation power to prove its work, which contributes to the large-scale use of ASIC and arouses new concerns. The use of this device breaks the original intention of Satoshi Nakamoto. TEE (Trustworthy Execution Environment) [164] provides a new method to solve this problem. One example of TEE is CPUs that support Intel SGX. It is proved that TEE can avoid the problem of uneven computing power, because only supported platform can be used for mining. The consensus algorithms using TEE functions can provide mining fairness, time security, and node identity certainty. PoL [164] utilizes a TEE platform's random number generation to choose a consensus leader. The benefits of this practice are little energy consumption, low latency transaction confirmation and equitably distribution mining. It is mainly composed of two functions, $PoLRound$ and $PoLmine$. These two functions also represent two stages of the algorithm. At the beginning of each round, participants prepare the TEE to mine on a specific chain by calling $PoLRound$. After $ROUND\_TIME$, the participant invocates $PoLMine$ to mine a new block. The $PoLMine$ function generates a random value in [0, 1). This value conforms to uniform distribution and is used to determine the winning block among blocks submitted by all participants. For the algorithm's security (liveness and persistence), in PoL, Milutinovic et al. also provides proof that a few nodes in the system can turn down the whole ledger records by fork. Given M as the majority population size, m as the minority population size, then for block t, the population luck conforms to the maximum of uniform random variables: $l_M(t) \overset{iid}{\sim} maxUniform(0,1)^M$, $l_m(t) \overset{iid}{\sim} maxUniform(0,1)^m$. After h blocks from a fork, the probability that the minority population wins is $Pr(L^{(h)\leq 0} \leq \min_{s>0} \mathbb{E}[e^{-sL^{(h)}}] = \min_{s>0} \prod_{t=1}^{h} \mathbb{E}[e^{-sl_M(t)}]\mathbb{E}[e^{sl_m(t)}] = \min_{s>0}(\mathbb{E}[e^{-sl_M(t)}]\mathbb{E}[e^{sl_m(t)}])^h$. Since $M > m$, it is obvious that the probability is less than 1. In other words, it is impossible for a few attackers to produce a branched chain that is more than most of the honest participants.

- PoC (Proof of Capacity, Proof of Space)

PoC is different from the above algorithms. Instead of competing computation power, it uses the available space on the hard disk. It means that the more hard disk space is provided, the more likely it is to get the right to account [165], [166]. The algorithm is completed by a prover and a verifier. The validation process consists of two stages. The first stage is the initialization phase that stores some useful data (the data is likely to be book material or some other media data). The verifier only needs to keep a small amount of information. In the second stage, the verifier will execute a program to judge whether to accept or deny the verification according to the data submitted by the prover. When verifier finishes the verification, it does not need to let the prover send the whole files, which will cause huge communication load. Usually, the verifier generates some queries to let the prover answer. If the prover follows the rules to provide storage

space, it can easily find the desired results. From a technical perspective, it needs to construct a directed acyclic graph. Each vertex in the graph contains a set of hash values. The verifier is supposed to find the right vertex and return the hash value at low communication costs, thus completing the verification. The interaction of two parties in the protocol can be expressed as $(out_V, out_P) \Leftarrow < V(in_V, P(in_P) > (in)$, where P denotes for prover, V for verifier, in_P and in_V for their local inputs respectively, out_P and out_V for their local outputs respectively.

SpaceMint [166] is slightly different from model [165] in performance. Dziembowski et al. [165] tended to regard the system as a state machine, while Park et al. [166] were more inclined to take it as an exchange. The nature of the two works is similar, while Dziembowski et al. [166] reduce the rate of interaction, thus the whole system has higher dispersion, better security. Both two models prove the security of PoC in detail, and the stability under different attacks is also analyzed. As early as 2014, Burstcoin began to use the PoSpace (PoC) algorithm, and Burstcoin was the first encrypted currency to use the PoC algorithm. A large number of small data generated during mining by the miners (ie., the certifier we mention before) are called "plot". These small data then are saved to disk. The number of plot determines the mining speed, and with more small data, it will be easier to find a block. However, because of the defects of the burstcoin algorithm, it is unable to resist the attack of time-space tradeoff, so it does not move towards the mainstream market. In 2015, the Spacemint [166] expanded the PoC theory on the basis of work [165]. Also, researchers from MIT, IST Austria, and Inria/ENS propose the Chia Network [167], which further improves the PoC algorithm. Readers are encouraged to find more details in [167].

- PoA (Proof of Activity)

PoA tries to extend PoW via PoS. Motivated by the threat of the infamous 51% attack in PoW, which could cause double-spending or PoW DoS attack, Bentov et al. [168] borrow the idea of PoS to improve the security and efficiency of PoW. The primary subroutine that PoA corporates is called *follow-the-satoshi*, whereby PoA transforms some pseudorandom values into a *satoshi* (a small unit of the cryptocurrency). The *satoshi* is randomly picked among all the *satoshi*s that have been minted thus far. The process of picking *satoshi* can be regarded as picking a pseudorandom stakeholder in a uniform fashion. There are five steps in PoA.

1) Each miner uses his or her computation power to generate an empty block header with only the address of the previous block, the miner's public address, and a nonce.
2) When the miner successfully generates the empty block header, he or she will broadcast this header to the whole network.
3) All the network nodes take the hash of this block header as data that deterministically derives N pseudorandom stakeholders.
4) Every online stakeholder checks the validity of the empty block header by comparing the hash of the previous block and the current difficulty. Then check whether the miner is one of the N lucky stakeholders of the block. When the Nth stakeholder finds the block derives from him or her, he or she will create a wrapped block with as many transactions as possible and extend the empty header with this wrapped block.
5) The Nth stakeholder broadcasts the block to the network, and all the other nodes in the network will validate this block after check. Then the block will be extended to the longest branch of the blockchain. It also needs to be mentioned that to alleviate the Tragedy of the Commons attacks, the fees from the transactions that Nth stakeholder collects will be shared by the miner and the N lucky stakeholders.

In addition to PoA, there are also other consensus mechanisms combining both PoS and PoW. Both TwinsCoin [169] and 2-hop blockchain [170] utilize PoW and PoS in parallel rather than use different algorithms in stages as PoA. The idea is to use physical resources (work) and virtual resources (stakes) to ensure the correctness and reliability of the ledger. In this mixed mechanism, the mining process is as follows:

1) The miner finds one valid PoW block;
2) The staker mines another PoS block;
3) It goes to step 1).

The PoW blocks and PoS blocks are mined alternatively and repeated, which means the PoW chain and the PoS chain are closely intertwined, as shown in Fig. 11. Apparently, it is not enough to deny the records of blockchain ledger only by bifurcation. It is also necessary to ensure that PoS blocks that are linked together can be rewritten for ledger records. Chepurnoy et al. [169] promoted this mechanism by increasing control difficulty in the process of generating two chains, making the whole system more robust. Denote $\widetilde{T_i}$ as PoS difficulty at *epoch* i, then the PoS difficulty at *epoch* $i+1$ is $\widetilde{T_{i+1}} = \frac{m_i R}{m} \widetilde{T_i}$, where m_i denotes the number of PoW blocks and m is the total number of PoS blocks and PoW blocks. Similarity, the PoW difficulty is $T_{i+1} = \frac{m t_i}{m_i t R} T_i$. From the block's view, it is a process that needs to be tried many times, as is shown in Fig. 12. The security of the system is also demonstrated in their work. It is also proved from the above-mentioned three aspects (i.e., CP, CQ, CG).

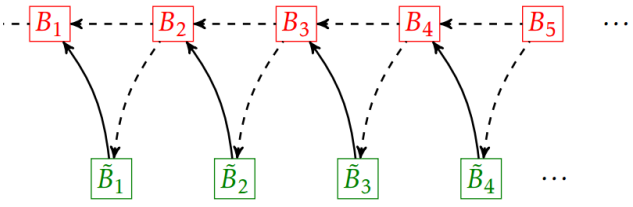


Fig. 11. A design combining PoW and PoS

- PoB (Proof of Burn)

PoB [5] states that PoW consumes too many real resources including human labor, hardware, and electricity. Therefore, it takes an alternative approach by "burning" coins in the system instead of costing real resources. In PoW, the more

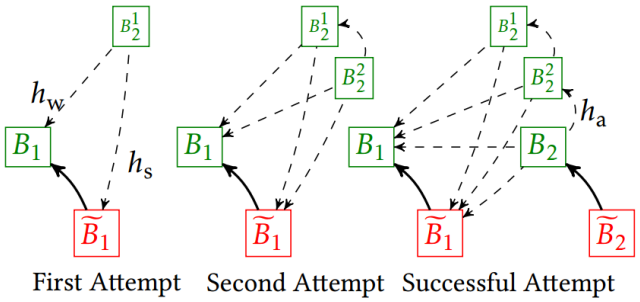


Fig. 12. Several attempts to mine PoW and PoS combined block [169]

a miner pays for computing devices (mining rigs) to solve the cryptographic puzzle, the higher chances that he or she could mine blocks. PoB works like virtual mining by burning virtual coins. Burning coins means sending coins to a predetermined address that is unspendable. When one miner in PoB burns coins, he or she buys virtual mining rigs. The more coins he or she burns, the more powerful the mining rigs will be, then the higher chances he or she will get to find out a block. In the real case, the mining rigs will be obsolete for frequent use and technology development, so miners often need to update the mining devices. In similar, the miners in PoB are also expected to burn coins periodically to keep competitive in mining blocks.

- PoI (Proof of Importance)

PoI is the blockchain consensus algorithm used by NEM [171]. The key idea of PoI is introducing the concept of importance to measure one account's ability to mine a block. The importance is determined by the amount of coins it has and the number of transactions made to and from this account. Motivated by computing the importance of nodes in graphs in areas of search, social network and others, NEM claims its core innovations is to use graph theoretic measures as a fundamental input into consensus. The transaction graph can be used to elucidate the importance of an account in PoI. NCDawareRank [172] is used to determine the salience or importance of a node in the network. Taking the vested balance in PoI as the "stake" in PoS, PoI shares great similarity with PoS. In PoS, an account should own more stakes to mine a block, while PoI considers both the transaction volume and trust. To mine a block, not only one should hold NEM but also actively carry out transactions. By applying PoI, NEM gains the advantages of energy efficiency and high transaction rate.

- PoET (Proof of Elapsed Time)

In order to improve the efficiency of PoW, PoET consensus algorithm in Sawtooth [173] offers another solution by exploiting the elapsed time that is actually random wait time created by the system. PoET is a lottery-like algorithm that meets fairness, investment, and verification. At a high-level, peers in PoET are stochastically elected to wait for a random amount of time and the peer who first finishes waiting will be elected as the leader for creating the new block. Two requirements need to be verified to ensure the peers in PoET act as expected. The first one is that the leader actually waits for a random time instead of a short time to win. The second is the leader indeed waits for the given wait time. To thwart these cheating acts, PoET uses Intel SGX as the trusted execution environment. There are three key functions in PoET. The first one is enclave, a protected area in an application's address space, which provides confidentiality and integrity. An enclave can act as a trusted function to assign a wait time for a request from the validator. The second one is "CheckTimer" that creates a timer for a transaction block that is guaranteed to be created by the enclave. The third one is "CheckTimer" used to verify whether the timer is created by the enclave.

E. BFT (Byzantine Fault Tolerance) and Variants

- PBFT (Practical Byzantine Fault Tolerance)

PBFT [138] was proposed by Castro and Liskov to tolerate Byzantine faults and made it even practical to be used in asynchronous environments while the previous algorithms could not. It is considered to be the first practical algorithm to achieve consensus when dealing with Byzantine faults. Particularly, there are three phases to achieve consensus in PBFT (i.e., *pre-prepare*, *prepare*, *commit*), as is shown in Fig. 13. In this figure, there are 4 replicas in which replica 0 is the primary and replica 3 is faulty, **C** is the client. As the primary, when replica 0 receives a request from client **C**, it starts the three-phase protocol. In *pre-prepare* phase, the primary assigns a sequence number to this request and multicasts the request to the other replicas (i.e., replica 1, 2, 3). When replica B and replica C receive this message, they respond it with *prepare* message and multicast the *prepare* message to other replicas. Then in *commit* phase, replica 0, 1, 2 multicast the *commit* message to others. Finally, replica 0, 1, 2 reply client **C** with a sequence number respectively. Assuming there are n replicas, in which f is the number of Byzantine faulty replicas, it is noteworthy that as long as $n \geq 3 \times f + 1$, the consensus can be achieved in PBFT. In other words, PBFT can offer both liveness and safety even when no more than $\lfloor \frac{n-1}{3} \rfloor$ are faulty simultaneously. Sukhwani et al. [174] investigated the performance of PBFT for networks with a large number of peers. Miller et al. [175] stated that PBFT critically relied on network timing factors, and were not suitable for real deployment. Therefore, they proposed a first practical asynchronous BFT protocol named HoneyBadgerBFT, which could support tens of thousands of TPS and more than a hundred nodes without network timing demands.

- Tendermint

Tendermint [176] [177] is an almost asynchronous BFT consensus protocol. It is optimized for PBFT and only requires two rounds of voting to reach consensus. Participants in Tendermint are called validators who take turns proposing blocks of transactions and voting on them. Blocks are committed in a chain, with one block at each height. A block may fail to commit for various reasons, so the consensus for a block may take several rounds. In each round, there are three steps (i.e., *Propose*, *Prevote*, *Precommit*). When more than 2/3 of precommit votes are received to achieve consensus in a round, the consensus for next height will start. An important concept in PoL is called PoLC (Proof of Lock Change), which

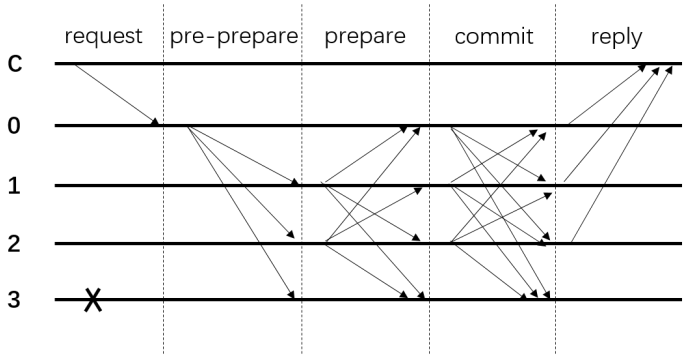


Fig. 13. Three phases in PBFT

represents prevote set exceeding 2/3 of total nodes for a block or null block in a particular height and round. In *Propose* phase, the nodes take turns to raise proposals. When non proposal nodes receive the proposal message, it turns into *Provote* phase. If there are locked blocks, it needs to collect the prevote for all of them. In *Prevote* phase, if nodes have locked blocks, a new PoLC for another block and satisfy $LastLockRound < PoLC - Round < CurrentRound$, then the locked block will be released. If the nodes still have locked blocks, then continually vote prevote for locked blocks, otherwise vote proposal or *nil* (null block). When more than 2/3 of votes for one block or *nil*, then it goes into *Precommit* phase. In *Precommit* phase, if a block has PoLC, then it will be locked and its LastLockRound will be set as the Current Round. Also, this block will be voted as precommit. If there are PoLC for nil, then unlock it and vote as precommit, otherwise keep its state as locked block and vote nil. When there are over 2/3 precommit votes for one block, then this block will be committed and come to consensus for the next height of block.

- Ripple

The Ripple Protocol Consensus Algorithm (RPCA) [34] is applied in Ripple. It utilizes collectively-trusted subnetworks within the larger network to achieve consensus for the Byzantine General Problem. In Ripple, Unique Node List (UNL) is a set of other servers maintained by each server, which plays an important role when a server makes queries to determine consensus. Only the votes from servers in UNL are considered when determining consensus. This is an obvious difference from many consensus algorithms. UNL represents a subset of the network that takes collective wisdom to make a consensus. The premise in RPCA is that each server trusts the other servers in UNL and believe they will not collude. RPCA proceeds in multiple rounds to achieve consensus. In each round, each server first collects as much as transactions to prepare for consensus and make them public in the form of "candidate set". Then each server makes a union of the candidate sets of the servers on its UNL, and vote for each transaction. According to the voting results, the transactions getting votes lower than a minimum percentage will be discarded or put into candidates set in the next consensus for the next ledger, while those who get enough votes will come to the next round. In the final round, all the transactions getting more than 80% from the UNL will be put into the ledger. Since each UNL represents a subset of the network, the final consensus is made by the whole network. Though Ripple could maintain robustness in the face of Byzantine failures, Todd [178] found there were some attacks against Ripple consensus. Armknecht et al. [179] stated the deployment of Ripple was not decentralized and Ripple Labs owned unconditional power to control the security of all Ripple transactions.

- Stellar

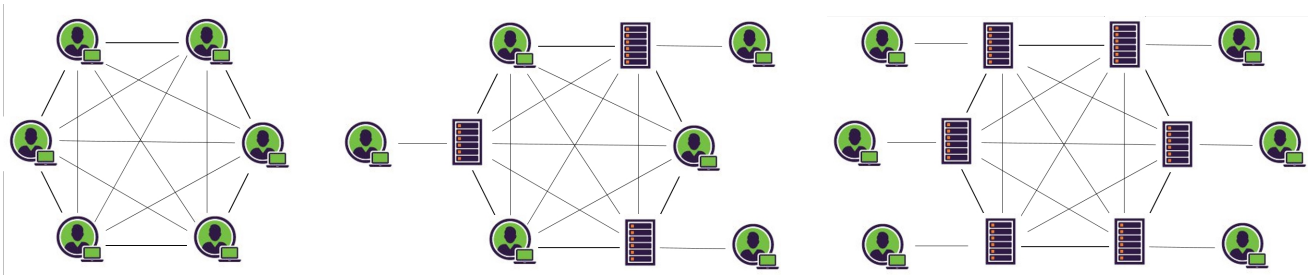
The Stellar Consensus Protocol (SCP) [180] is a construction for federated Byzantine agreement (FBA). SCP is considered to be the first provably safe consensus mechanism to simultaneously enjoy four key properties: Decentralized control, Low latency, Flexible trust, and Asymptotic security. Asymptotic security means that the safety of SCP rests on digital signatures and hash families whose parameters can realistically be tuned to protect against adversaries with unimaginably vast computing power. SCP borrows the concept of quorum in the distributed system to make a consensus. A quorum is a set of nodes sufficient to reach an agreement. SCP introduces the concept of a quorum slice, which is the subset of a quorum that can convince one particular node of agreement. Quorums that overlap share common nodes, which is called quorum intersection, otherwise they are called disjoint quorums. Disjoint quorums could agree on contradictory statements, which will undermine the consensus.

The voting process of SCP includes three phases: *vote*, *accept* and *confirm*. Take an example of lunchtime consensus that a group of people agree on what to eat during lunchtime. In initial voting, one person chooses pizza and keeps it open. Then because of quorum intersection, quorum slices influence one another. If there are other ones in his or her quorum slice, they can reject other food options. In *accept* phase, a *v-blocking* set of nodes plays an important role because it can block actions in all quorums that contain this person, causing he or her to accept other option. A *v-blocking* set of nodes is the set contains at least one node of this person's slices. If all in a quorum vote for pizza, then pizza is ratified by this quorum. Then come to the *confirm* phase. Confirmation is the final step to achieve system-wide agreement. Nodes exchanges confirmation messages in this phase. Only after sufficient messages are delivered and processed on a statement, will the subsequent event accept the statement, then the agreement can be achieved.

F. Consensus algorithms comparison

In the above sections, we describe the current popular consensus algorithms in blockchain. In this section, a comparison of these consensus algorithms is made, as is shown in table II. Some items in this table can be explained as follows:

- Blockchain type. According to the decentralized degree, blockchain can be classified into three types: public blockchain, consortium blockchain, and private blockchain. Public blockchain has been the most widely used blockchain. It has four characteristics:



(a) Public blockchain (b) Consortium blockchain (c) Private blockchain

Fig. 14. The three blockchain types

1) **Easy to access**. Everyone who has a computer connected to the Internet can access this blockchain.
2) **Transparency**. The transactions are public and transparent for everyone in the system.
3) **Anonymity**. All the transactions are operated anonymously so that the privacy of the users is well protected.
4) **Free of control**. Anyone including the developer cannot control or tamper the data in the system.

Bitcoin is the most representative public blockchain application. The consortium blockchain is managed by some organizations or institutes, who grant people to participate in. At a cost of some decentralization, consortium blockchain can provide the efficiency and security of public blockchain while keeping some central control, monitor and safeguarding. Hyperledger is one of the most famous consortium blockchain applications. In private blockchain, the permissions are even limited and retained in one individual or an organization. Therefore, in some cases, some rules or even data can be modified or tampered by the authority. Compared with public blockchain and consortium blockchain, private blockchain gains the advantages in high speed and throughput. Private blockchain is often applied for individual company. One example could be YobiChain [181], a private blockchain ecosystem preloaded with database, web & FTP servers and a simple blockchain application. The three blockchain types are shown in Fig. 14.

- Transaction finality. Transaction finality refers to a property that whether a block has been added to the blockchain will be removed from it later. For example, in PoW, forks with the same length in each chain may exist simultaneously because of computational competence for incentives. Therefore, the transaction finality of PoW is probabilistic.

G. Scalability

Scalability is an important consideration when applying a blockchain into practice. Blockchain scalability often involves the transaction throughput (i.e., the maximum rate when processing transactions), latency (i.e., the time need to confirm a transaction) and the number of supporting nodes. As the most popular blockchains, Bitcoin and Ethereum are mainly used for investment and cannot support mainstream transaction usage for the scalability problem. The scalability of a blockchain mainly depends on the consensus mechanism it applies.

The scalability issue has been widely discussed. Bitcoin has long been known as poor scalability, whose processing capacity is between 3.3 and 7 transactions per second (TPS) [16]. In [16], Croman et al. pointed out that the increasing popularity of blockchain-based cryptocurrencies made scalability a primary and urgent concern. In their work, they not only discussed the transaction throughput, but also the latency. As a representative blockchain, it is pointed out in their work that Bitcoin can only support 7 TPS and takes 10 minutes or longer to confirm a transaction. By contrast, Visa can process 2000 TPS on average, with a peak rate of 56,000 TPS. Their findings suggested reparameterization of block size and intervals should be viewed only as a first increment toward achieving next-generation, high-load blockchain protocol. More specifically, to ensure more than 90% of nodes in the current overlay network have sufficient throughput, the block size should not exceed than 4MB and the block interval should not be smaller than 12s.

The community has several propositions to scale blockchain, which can be listed as follows. Jordi and Cristina [182] also stated that the low capacity in transaction throughput had become a main drawback for Bitcoin to be a global payment system. In their study, they provided a comprehensive description of the most relevant scalability solutions proposed for the bitcoin system. The solutions included tuning Bitcoin protocol parameters, the segregated witness approach. Karame et al. [183], [184] also proposed schemes to increase the Bitcoin transaction rate. Malavolta et al. [185] stated that permissionless blockchains (e.g., Bitcoin) were limited in transaction throughput and latency. To address this issue, off-chain payment channels had been proposed to be combined in Payment Channel Network (PCN). However, this approach would raise a serious privacy concern. Therefore, they proposed two payment protocols to tackle the privacy and concurrency issues. Forte et al. [186] pointed out that the unsustainability of mining in massive large-scale blockchain system. They found Bitcoin-like cryptocurrency systems were potentially not energy efficient in a global scale.

TABLE II. COMPARISON BETWEEN THE CURRENT POPULAR CONSENSUS ALGORITHMS

	PoW	PoS	DPoS	PBFT	Ripple	Stellar	Tendermint
Year	1999	2012	2014	1999	2012	2015	2014
Blockchain type	public	public	consortium	consortium	consortium	consortium	consortium
Key concepts	hashpower	stake, coin age	delegate, stake	replica	UNL	quorum	validator
Transaction latency	high	low	low	low	low	low	low
Transaction throughput	7 tps	100 tps	100,000 tps	100s tps	1500 tps	3000+ tps	10,000tps
Transaction finality	probabilistic	probabilistic	probabilistic	immediate	immediate	immediate	immediate
Need mining?	yes	yes	yes	no	no	no	no
Token needed?	yes	yes	yes	no	no	no	no
Energy efficient?	no	yes	yes	yes	yes	yes	yes
Adversary tolerance	$< 50\%$ hashpower	$< 50\%$ stake owner	$< 50\%$ validator	$\leq \lfloor \frac{n-1}{3} \rfloor$	$\leq \lfloor \frac{n-1}{5} \rfloor$	It depends	$\leq \lfloor \frac{n-1}{3} \rfloor$
Typical applications	Bitcoin, Ethereum, Peercoin, Litecoin, Dogecoin	Ethereum, Peercoin, NXT, NVC, Cosmos Coin	Bitshares, ARK	Hyperledger	Ripple	Stellar	Cosmos Network, Ethermint

1) Adjusting block size: Croman et al. [16] suggested that reparameterization of block size and intervals should be viewed only as a first increment toward achieving next-generation, high-load blockchain protocol. More specifically, to ensure more than 90% of nodes in the current overlay network have sufficient throughput, the block size should not exceed than 4MB and the block interval should not be smaller than 12s. Jordi and Cristina [182] held that the major throughput limit was the maximum size of blocks, and suggested tuning protocol parameters to increase Bitcoin's scalability. Many Bitcoin Improvement Proposals (BIPs) [187] have been proposed to increase the scalability of Bitcoin, as is shown in Table III.

TABLE III. THE BIPS RELATED WITH ADJUSTING BLOCK SIZE

BIP	Proposal	Author	Status
100	Initiate the maximum block size as 1 MB, then dynamically change it encoding a proposed value and a vote.	Jeff Garzik, Tome Harding, Dagur Valberg Johannsson	Draft
101	Replace fixed maximum block size 1 MB with a new one growing over time at a predictable rate.	Gavin Andresen	Withdraw
102	Change the block size form 1 MB to 2 MB.	Jeff Garzik	Draft
103	Replace the block size by a function, applied to the median of the timestamps of the previous 11 blocks.	Pieter Wuille	Draft
104	Dynamically adjust the max block size with the target of keeping blocks 75 % full, base on the average block size of the previous 2016 blocks.	t.khan	Draft
105	When a miner creates a block, they may vote to increase or decrease the block size by a maximum of 10 % of the current limit size.	BtcDrak	Draft
106	Replace the fixed maximum block size with a dynamically controlled one that may increase or decrease with difficulty change depending on various network factors.	Upal Chakraborty	Draft
107	Changing block size limit based on transaction volume.	Washington Y. Sanchez	Draft
109	Changing block size limit from 1 MB to 2 MB.	Gavin Andresen	Rejected

2) Sharding: Sharding is a technique used to partition large databases into smaller, faster, more easily managed parts called data shards [188]. Aeternity [189] presents a blockchain architecture with high scalability. It takes three designs to increase its scalability: sharding trees, light clients, state channels and parallelism. First, it can shard the state data across many nodes to lessen each node's load. Merkle trees are used to prove the state. Light clients do not download the entire block but keep a hash value and only download the headers of the blocks. Status channels have immense throughput and most transactions inside them are never executed or even recorded on the blockchain. Status channels can also facilitate the parallel processing of transactions, thus further increase Aeternity scalability. Sharding protocols have also been widely applied into many distributed databases, such as MongoDB [190], MySQL, Spanner. Mcconaghy et al. [191] described BigchainDB, a scalable blockchain database, to fill the gap in the decentralization ecosystem. Luu et al. [192] proposed the first distributed agreement protocol for permission-less blockchain called ELASTICO, which can tolerate one-fourth byzantine adversaries of the total computational power. Croman et al. [16] considered sharding the task of consensus among concurrent operating sets of nodes. Gencer et al. [193] designed a sharded blockchain protocol called Aspen to securely scale the increasing number of services.

3) Off-chain payment: Lightning network [45] is capable of dealing with millions to billions of transactions TPS, which take the strategy of transacting and settling off-chain. It takes bidirectional payment channels that is a ledger entry signed by both parties to accomplish the transactions. Since the transaction can be conducted off-chain without delegation of ownership and trust, users are allowed to launch unlimited transactions between other devices. Decker and Wattenhofer [194] exploited off-blockchain transactions to create long-lived channels for transfers, which alleviated the burden of Bitcoin network.

H. Attacks against the consensus mechanisms

Due to the prevalence of blockchain and its widespread applications, there are various attacks against the consensus mechanisms. In this part, we summarize these attacks and analyze the characteristics of these attacks.

- Double-spending Attack. The double-spending attack is a potential risk in digital cash scheme for the duplicability. In the real world, once spent, paper money or coin cannot be reused, while it is possible in the digital world. Because of the reproducible nature of digital assets, the transfer of digital value cannot be carried out as real ones. Adversaries are likely to send one digital coin to multiple parties to gain more benefits.
  The double-spending attack has been discussed in many literatures. Nakamoto solves the double-spending problem by using a peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions [2], which is the prototype of bitcoin.
  Karame and Androulaki [183], [184] found that to prevent double-spending, Bitcoin required too much time to verify a transaction, thus inappropriate for fast payment. Also, they found the recommendations proposed by Bitcoin developers were not always effective in detecting double-spending attack. In this study, they proposed remedies to the existing Bitcoin implementation to ensure detection of double-spending attack for fast payment. However, Gervais et al. [195] found ways to circumvent the remedies and double-spend fast payment without losing money. The authors further exposed some solutions to deter the misbehavior and give a lower-bound on the wait time to secure fast payment. Bissias et al. [139] analyzed the double-spending attacks with and without eclipse attack, and quantified the attack parameters to determine the attack success. They found the security of a transaction against double-spending increases logarithmically with the depth of the block, which made it easier by increasing the potential incentive but difficult by promoting PoW. Lajoie et al. [196] stated that to prevent double-spending attacks and forks in Bitcoin, the fundamental key was to impose synchronization constraints. Prinzon and Rocha [197] contributed two double-spending attack models for Bitcoin. Through their experiment, they concluded that even when an attacker had some reasonable time advantages, it was still unlikely that double-spending attacks are practical. Courtois [198] argued the built-in properties (e.g., the longest chain rule) were not as brilliant as they were claimed. He mounted 51 % attacks to double-spend the cryptocurrencies.
- Eclipse Attack. In an eclipse attack [118], the attacker controls all the victims' incoming and outgoing connections to isolate the victim from the other peers in the P2P network. By virtue of this isolation, the attacker can exploit the victim to attack bitcoin's mining and consensus system, which includes double-spending, selfish mining and other attacks. In their paper, Heilman et al. proposed two countermeasures that included disabling incoming connection and choosing outgoing connections to peers in a whitelist.
- Selfish Mining Attack. In selfish mining attack [199], the attacker can release mined block lately and control the length of blockchain to surpass the original length. It will make the honest miner work in vain, thus forcing honest miners to join the dishonest camp. In Eyal and Sirer's work [199], they proved that the overwhelming majority of honest miners were not enough to ensure the security of the bitcoin protocol. Selfish mining could be exploited by a minority pool to obtain more revenues than the pool's ratio of the total mining power. The key idea behind this strategy is that a pool keeps its discovered blocks private and continues to mine more blocks to gain advantages, while other honest pools keep wasting their computing power to mine on the public chain. The consequence of this strategy is that more honest miners are incentivized to join the selfish parties to gain more benefits, which makes Bitcoin system cease to be a decentralized currency. Bitcoin utilizes two incentives to motivate miners, which are block rewards and transaction fees. Since transaction fees will play the major incentive role in Bitcoin when it is too difficult to mine a block for rewards, Carlsten et al. [40] provided their insights about this particular future scenario when only transaction fees existed. They find only applying transaction fees will incur security issues, such as selfish mining.
  To thwart the selfish mining attack, the authors also suggest infiltrating selfish pools and other countermeasures. Sapirshtein et al. [200] further investigated the profit threshold in selfish mining and achieved more findings different from that in [199]. In their work, they evaluated the countermeasures for selfish mining attack but showed they were less effective as expected. The authors also discussed the iteration between selfish mining attacks and double spending attacks, then they demonstrated how a selfish mining attacker could launch double-spending attacks without extra costs. Gobel et al. [201] used a simplified Markov model to track the status of honest miners and dishonest miners. Then, the spatial Poisson process model was used to study the parameter γ mentioned in Eyal and Sirer's work [199]. Finally, discrete event simulation was used to study the behavior of miners in the network. Nayak et al. [119] investigated the attack strategies to increase revenues of the attacker. In their work, they found selfish mining was not optimal for a large parameter space and suggested "stubborn mining" strategy, which was a selfish-mining-style attack but could gain more revenues (up to 25%) than traditional selfish mining attack. The "stubborn mining" attack includes a strategy called trail-stubbornness in which the attacker keeps mining on the private fork even the public fork is ahead. They claim that a trail-stubborn strategy in some cases can gain 13% revenue compared with a non-trail-stubborn counterpart.
- Block withholding attack. Rosenfeld [35] discussed the

abnormal situation that miners just withheld the blocks they found but did not or delayed to submit them. Based on whether the attacker submitted the discovered block, the author classified block withholding attacks into two kinds of attacks, "sabotage" and "lie in wait". In "sabotage" attack, the attacker never submits the blocks, while the blocks will be delayed to submit in "lie in wait". Both attacks will jeopardize the blockchain system. Courtois and Bahack [202] further studied withholding attack and proposed a new concrete and practical block withholding attack to maximize the advantage gained by rogue miners. To thwart block withholding attack, the authors held that the only defense was involving people that trusted each other in the pool and the manager should dissolve and close the pool once earning is less than expected. Tosh et al. [203] modeled the block withholding attack in a blockchain cloud considering distinct pool reward. The results demonstrated the attacker's access to extra computational power could disrupt the honest mining operation in blockchain cloud. It was to be mentioned that selfish mining attack was also a kind of block withholding attack. Kwon et al. [204] proposed an attack, which was called fork after withholding (FAW). The attacker in FAW could gain equal or more rewards than block withholding (BWH) attack and it was more usable.

- Other Attacks.
  Finney attack [205] is proposed by Finney in the bitcoin community. The Finney attack proceeds as three steps: 1. the attacker tries to include a transaction that deposits some of his own coins to himself without broadcasting it; 2. when he minds a block, he sends the deposited coins to a merchant for real goods still without broadcasting; 3. as the merchant accepts the payment, he broadcasts the block to override the unconfirmed payment to the merchant. In Finney attack, the attacker gets the goods or service without paying. Vector76 attack [206] is another attack that happens even under one confirmation. Sompolinsky and Zohar [145] further discuss Finney attack and Vector76 attacks in detail. Among them, the attacker's attack strategy (i.e., the behavior in different states) and the honest miner's acceptance strategy are all defined as corresponding function mappings, and four different versions of security guarantees are proposed.
  Goldfinger attack [148] is another attack derived from 51 % attack to destroy the Bitcoin economy with the aim of achieving utility outside the Bitcoin economy. The authors hold there are many motivations for Goldfinger attack. For example, the government might want to block Bitcoin transactions to deter money laundering. Bradbury [207] also pointed out the code-based attack in which the attacker might exploit the incomplete feature in the source code. Miller [208] proposed an attack strategy called feather-fork, in which a miner deliberately blacklisted some blocks they disliked. He held that by refusing to build on this block just for a short time, it created an incentive for other miners to enforce the blacklist as well, thus achieving the conspiracy of blocking a transaction.
  Balance attack is identified by Natoli and Gramoli [209], [210], which can be used to target forkable blockchain systems. The novelty of this attack includes delaying network communication among multiple subgroups of nodes with balanced mining capabilities. In another work [211], they introduce the blockchain anomaly, which prevents users of mainstream blockchain systems from performing conditional transactions, and describes a possible way to avoid exceptions by writing smart contracts instead of transactions.
  Another attack is mounted to destroy an open game pool by pretended participation, but not to share its work proof. Eyal [212] defined and analyzed a game in which the pool used some of their participants to infiltrate the participants in other pools to perform the above attacks. He also found that there was an equilibrium that constituted the tragedy of the commons where pools attacked each other. To decide whether or not to attack, two pools would fall in the *the miner's dilemma*. The analyses showed that the game was played every day by active bitcoin pools, and it was clear that the pools would not choose to attack. If this balance broke, the income of the open pool might be reduced, making them unattractive to the participants.

I. Related Countermeasures

This part discusses the defending strategies for the above-mentioned attacks. The GHOST (Greedy Heaviest Observed Subtree) rule [144] is a famous one, and its variants have been applied to the Ethernet workshop project. In Sompolinsky and Zohar's work [144], they first analyzed the impact of bitcoin security in the case of having higher throughput. As is shown in Fig. 15, when the block size and block creation rate increase, though the throughput of blockchain can be promoted, the risk of occurring forks in block tree also increases, which further reduces the security of blockchain. The GHOST protocol can solve this security problem by aiming to replace the principle of the longest chain with the heaviest sub-tree, as is shown in 16. Through modifying the bitcoin blockchain node construction and organization mod, even the computing power of dishonest node reaches 50%, and the security of the entire system can be guaranteed. In Fig. 16, even the attack secretly creates a chain to be the longest chain, it still cannot outweigh the main chain constructed by GHOST protocol.

Natoli and Gramoli [209], [210] quantified the tradeoff between the network delay and mining power of the attacker needed to double-spend in the GHOST protocol with high probability. Kiayias and Panagiotakos [146] showed that GHOST actually worsened the "chain quality" (part of the honest block in the chain). Therefore, it was a new framework based on work [144] for analyzing blockchain protocols established by trees instead of chains. This framework could make a unified description of the GHOST protocol in Sompolinsky and Zohar's work [144] and the Nakamoto's bitcoin framework agreement. In the PoW mechanism of the bitcoin system, the cost of mining will eventually turn to zero as time goes on.

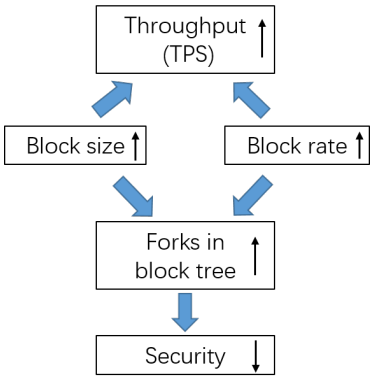


Fig. 15. Bitcoin security under high throughput

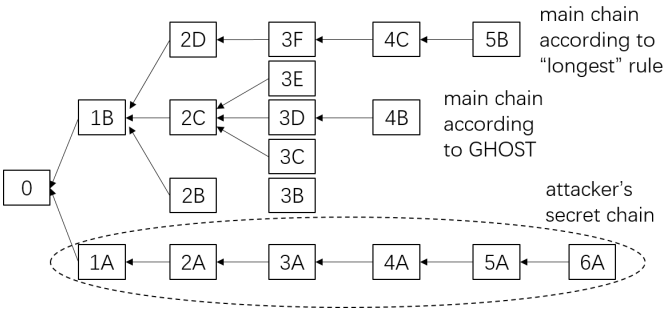


Fig. 16. The heaviest sub-tree design in GHOST protocol

Biryukov et al. [213] found that fast verification demand in PoW made a prey for ASICS-, GPU-, and botnet-equipped users. Therefore, they proposed an asymmetric PoW called Equihash, which required a lot of memory space.

Babaioff et al. [214] discussed information dissemination and Sybil attacks, and introduced an incentive mechanism that enabled the miners to spread the transactions they knew. In the work of Garay et al. [215], the precomputation assumption of no confrontation in their previous work [143] was removed. The authors showed a similar bitcoin blockchain protocol of bootstrapped-like bitcoin blockchain protocol, which depended on PoW that generated blocks from scratch in the case of pre-computation.

Gervais et al. [140] proposed a quantitative framework to analyze the security and performance of various consensus and network parameters of PoW blockchain. They also devised optimum adversarial strategies for double-spending and selfish mining while taking real world constraints into consideration. The security framework is shown in Fig. 17. In their work, they showed the higher the block reward of a blockchain, the more resilient it would be against double-spending attack. Also, they stated that the current PoW blockchain could attain transaction throughput more than 60 TPS (the current throughput in Bitcoin is 7 TPS) without compromising the security of blockchain system. They also took the impacts of eclipse attacks into consideration.

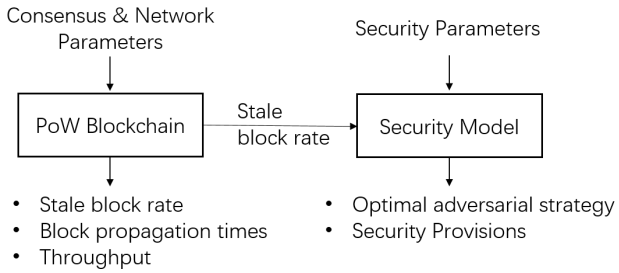


Fig. 17. The quantitative framework for PoW blockchain security analysis

V. Applications

Blockchain has been widely applied in various fields. In this section, we will introduce these blockchain applications. Generally, we first describe the blockchain extensions that some applications built on, then we introduce the general blockchain applications.

A. Extension

1) Smart contract: Smart contracts are systems that automatically move digital assets according to pre-specified rules [216]. With the widespread deployment of blockchain, smart contract has got a wide attention. The number of smart contract projects in Github has been reaching 445 units [10]. In Bartoletti and Pompianu's work [10], they collect 834 smart contracts from blockchains of Bitcoin and Ethereum, then classify them into five categories: financial, notary, game, wallet, and library.

Rodrigues et al. [11] utilized blockchain and smart contracts to share information to thwart DDoS attacks. Norta [217] formalized a smart-contracting setup lifecycle for Decentralized Autonomous Organizations (DAOs) negotiation phase. Bogner et al. [12] designed a Decentralised APP (DAPP) for the sharing objects based on a smart contract on the Ethereum blockchain. The smart contract enables users to register and rent objects without a Trust Third Party (TPP), while protecting the users' personal information. Mccorry et al. [13] presented Open Vote Network, a decentralized and self-tallying internet voting protocol written as a smart contract for Ethereum, while keeping maximum voter privacy. Yasin and Liu [218] proposed a smart contract management framework referring to personal online ratings based on the aggregated digital identity. Alharby and Moorsel [219] conducted a mapping study of smart contract related papers and find two-thirds of them focused on smart contract codifying, security, privacy, and performance issues, while the others focused on applications or else. They also pointed out five research gaps of smart contract: scalability issues, a limited number of researches on other blockchains using smart contract except for Ethereum, a small number of smart contract applications, lack of high-quality peer-reviewed research on smart contracts. Idelberger et al. [220] held that logic-based languages were possible interesting alternatives to procedural languages in programming smart contracts in blockchain system. In their work, they provided insights on

how to use logic-based smart contracts in combinations with blockchain systems. Searching encrypted data is a desirable technique in outsourcing where the servers are responsible for data verification. Noticing that there is no general verification mechanism suitable for all search schemes. Hu et al. [221] proposed a decentralized privacy-preserving search scheme by replacing the potential malicious central server with smart contract. Chen etc. [222] held that blockchain's lack of confidentiality and poor performance could impede the availability and security assurance of blockchain. Therefore, they proposed Ekiden, a system capable of performing concurrent, off-chain execution of smart contracts within TEE-backed compute nodes, while yielding high performance, low cost and confidential for sensitive data.

Security is of great importance to smart contracts in blockchain. In order to analyze and verify the runtime safety and functional correctness of Ethereum smart contracts, Bhargavan et al. [223] translated these contracts into $F^\star$ that is a programming language aimed at program verification. Zhang et al. [224] held that trustworthy data feeds supporting a wide range of data requests would be critical to smart contract ecosystems. Therefore, they proposed Town Crier, an authenticated data feed system. Kosba et al. [14] stated the existing systems lacked transactional privacy, and they proposed *Hawk* to enable programmer to write private smart contract without implementing cryptography. Juels et al. [225] evaluated the risk of smart contracts in fueling new criminal ecosystems. The criminal threats included confidential information leakage, crypto key theft, and various real world crimes. Luu et al. [9] investigated the security of running smart contracts and find several new security issues. They also built a symbolic execution tool to find potential bugs, which was called OYENTE. Krupp and Rossow [226] found security vulnerabilities were tightly intertwined with financial gain in smart contract, thus creating an opportune setting for attackers. Particularly, they proposed a method to find vulnerable execution traces in a smart contract and created symbolic execution to create an exploit.

2) Sidechains: Different from forks that update the existing blockchain, sidechains allow the developer to attach new features to the main chain in a separate chain. Sidechains enable interchangeability of digital assets between them and their parent blockchain. In 2014, Back et al. proposed *pegged sidechains* [15] to enable bitcoins and other ledger assets to be transferred between multiple blockchains. *Pegged sidechains* gain two distinguished merits: 1. Avoiding the liquidity shortages and market fluctuations associated with new currencies; 2. Though interconnected with the main blockchain, they are isolated, which means even one sidechain crashes, the damage can only be confined to the sidechain itself. Also, the sidechain technology will not hinder the technical and economic innovation of other chains.

Lesigns [227] designed a 2-way peg (2WP) protocol that allowed transfers of a cryptocurrency between main blockchain and a second blockchain with low third-party trust. Maxwell [228] utilized sidechain to bring new elements to Bitcoin. These new elements included confidential transactions, asset insurance, segregated witness, script enhancements, etc. Take "confidential transactions" as an example. "Confidential transactions" was a new feature of sidechains that ensured that the amount of transferred cryptocurrency could be only visible on the transaction parties [229]. This new feature could improve the privacy of blockchain transaction that exploited the pseudonym to achieve anonymity. Hueber [230] asserted that blockchain technology would become the keystone of many electronic markets and exploited sidechains to model a cryptocurrency regime. Noizat [231] designed a blockchain voting system and suggested using a sidechain as a precautionary option to generate particular public keys. Dilley et al. [232] improved sidechains with strong federations. In a federated sidechain, members of the federation serve as protocol adapters between the main chain and the sidechain.

There are also three technical challenges by introducing sidechains, which is pointed out in [16]. The first challenge is the inter-chains mining power coordination. Merged mining techniques allow separate chains to share their mining power. If there is no coordination, the maintenance of sidechains will dilute the mining power in the system, which renders the individual chains vulnerable. The second is that inter-chains transaction will put more pressure on the main chain, resulting in an adverse impact on the scalability. The third one is that transactions between chains will incur high latency.

B. Blockchain x.0

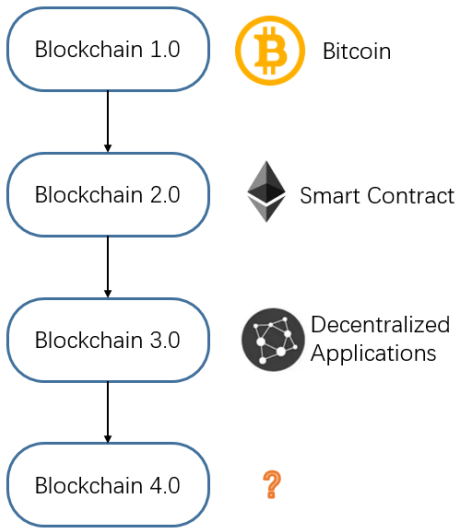


Fig. 18. The development of blockchain

Blockchain technology develops from blockchain 1.0, blockchain 2.0, to blockchain 3.0, as is shown in Fig. 18. The blockchain 1.0 is represented by virtual currency, such as bitcoin. The application of blockchain 1.0 is related to currency, such as currency transfer, currency exchange, and payment systems. Blockchain 2.0 can be represented by the decentralized applications written by smart contracts. Blockchain 2.0 applications cover other financial fields, equity, bonds, and credit. Blockchain 3.0 further extends the field of blockchain

applications beyond the financial industry, covers all aspects of human social life, and provides self-certification of information in various social activities, without relying on a third party or institution to gain trust or establish credit and share information, such as health care, intellectual property, internet of things, social management, charity and public welfare. There are also reports that blockchain 4.0 is coming [233], with some improvements of blockchain 3.0. However, there are a few real blockchain 4.0 applications and the characteristic of blockchain 4.0 is not clear. It is to be mentioned that Zhang and Jacobsen [234] claimed they gave a taxonomy of blockchain applications based on blockchain 1.0, 2.0 and 3.0. However, they give a limited description of these applications.

1) Blockchain 1.0 - Bitcoin: Bitcoin is the earliest decentralized cryptocurrency that was released as an open source software in 2009. By introducing a new distributed public ledger, users are allowed to spend bitcoin for P2P transactions while avoiding the double-spending issue of electronic payments. The transaction does not need to be overseen by any authority or server, but the decentralized network itself determines whether the payment is reasonable or not. Cryptocurrencies or token are created by mining blocks in a part of blockchains. For example, bitcoin is built directly on the Bitcoin blockchain mining. One can also choose to fork the Bitcoin blockchain and create new tokens including ZCash, Litecoin, Monero. The other way is to build a new blockchain and mine tokens on it, such as Ethereum. The token on the Ethereum blockchain is ETH.

In 2011, Litecoin appeared as an early alternative to Bitcoin. By 2011, mining Bitcoin would require more and more specialized and expensive hardware, making it difficult for ordinary people to mine Bitcoin. Litecoin is basically the same in technology as Bitcoin, but it is more lightweight. The Litecoin algorithm attempts to allow everyone with an ordinary computer to participate in the mining process. At present, other cryptocurrencies have taken away some of Litecoin's market share, but Litecoin still has early advantages and strong network effects.

Ripple Networks is a global settlement network for different currencies or value entities, such as the U.S. dollar, euro, pound, bitcoin and flight mileage, etc. Ripple tries to achieve a flexible money flow system, with its core as the debt relationship. Ripple network can serve as an intermediate to help two users complete the exchange of different value entities. In this way, the user only needs to establish a trust relationship with the intermediate transfer network. There is no need for users to establish a trust relationship with each other, thus greatly improving the value transfer efficiency. Specifically, a user attempting to finish a transfer deal needs to convert his or her money into XRP (i.e., the cryptocurrency of Ripple), then XRP will be converted into the intended currency with same value. Although Ripple can be traded on the cryptocurrency market, it is clear that the fundamental role of XRP is to assist the circulation of money in the global system.

Other digital currencies also gain their own features. For example, Dash [235] is a digital currency that supports instant transactions and protects user privacy. It is based on bitcoin and has a unique double-layered network that includes both miners and master nodes. It improves Bitcoin in two major aspects: transaction speed and anonymity. Dash's instant payment technology allows transactions to be completed almost instantaneously and uses currency blending techniques to ensure the privacy of transactions.

2) Blockchain 2.0 - Smart contract: As the representative of blockchain 2.0, smart contracts broaden the functionality of blockchain to a large degree, which is introduced in the above parts. Ethereum first realizes smart contracts, and publishes the whitepaper: *A Next-Generation Smart Contract and Decentralized Application Platform* [236]. It has been committed to making Ethereum the best platform for smart contracts. The application is running as programmed, and there is no possibility of downtime, censorship, fraud, or third-party human intervention. As a result, with Bitcoin leading the blockchain, Ethereum is reviving smart contracts. In the Ethereum blockchain, people can also mine for ETH. Using the logic of contracting, it can distribute ETH to other accounts and other smart contracts.

Ethereum can be referred to as a series of protocols that define a decentralized application platform. The core of this platform is the Ethereum Virtual Machine ("EVM"), which can perform the coding of arbitrary complex algorithms. Ethereum platform itself has no characteristics and no value. Similar to a programming language, its value is decided by developers. Ethereum is suitable for building applications that automatically interact directly between peers and peers or that facilitate team coordination activities across the network, such as the coordination of P2P applications. Also, it is applicable for developing automation of complex financial contracts. Bitcoin allows individuals to exchange currencies without relying on financial institutions, banks, or government, while the impact of Ethereum may be even more profound. In theory, any complex financial activity or transaction can be automatically and reliably coded in Ethereum. In addition to financial applications, any application scenarios that require high trust, security, and durability, such as asset registration, voting, management, and the Internet of Things, will be heavily influenced by the Ethereum platform.

Although digital currency represented by Bitcoin has achieved great success, Bitcoin's public chain cannot overcome its inherent problems. For example, the transaction efficiency is low, and blocks are not finalized. These problems make ETH and other public chains unable to meet the requirements of most commercial applications. The Hyperledger project aims to create a transparent, open, and decentralized distributed ledger project as an open source specification and standard for blockchain technology. It allows more applications to be built on blockchain technology. The goal of this project is to develop a cross-industry open standard and source code development library. It allows companies to create customized distributed ledger solutions by applying blockchain technology. At present, it mainly includes three major projects: Fabric [237], SawTooth [238] and Iroha [239]. The entire community is mainly managed by technical committees, management boards, and the Linux Foundation. The emergence of the project actually announces that blockchain technology is no longer an application scenario that is only for the "social

experiment" nature. It has been formally approved by mainstream organizations and enterprise markets. To promote the community size, Hyperledger has proposed and implemented complete rights management, innovative consensus algorithms, pluggable and extensible frameworks. These practices will have profound impacts on the development of blockchain technologies and the industries.

3) Blockchain 3.0 - Decentralized Applications: With the development of blockchain technology, it has been evolved to blockchain 3.0 era that witnesses multitudes of blockchain applications in various fields. The great potential of blockchain has aroused huge impacts on both the industry and academia. A lot of blockchain applications has been put into practice and achieve great success, such as the numerous kinds of altcoin in the financial market. In academia, researchers also make lots of attempts to apply blockchain in more domains. To start this part, a similar work to this part needs to be mentioned. Holub and Johnson [24] claimed they have a comprehensive research of the literature from an original sample of 13,507 results. They categorized and mapped a final sample of 1,206 sources across six disciplines, which included technology, economical finance, accounting, tax, and regulation. However, their work focused more on bitcoin and the paper numbers in each category, thus ignoring many important details. In this part, we overcome these defects and explore blockchain applications in different areas, which is shown in Fig. 19.

Market. Zhao et al. [240] believed the widespread adoption in finance and other business sectors would contribute to many business innovations as well as many research opportunities. Mendling et al. [241] outlined the challenges and opportunities of blockchain for business process management (BPM) to simulate researches utilizing blockchain. Their work reflected that not only blockchain could be used in the established BPM but also beyond it. They also discussed the blockchain technology and BPM capabilities in areas including strategy, governance, information technology, people, and culture. Prybila et al. [242] exploited blockchain to enable a seamless execution monitoring and verification of choreographies, while preserving the anonymity and independence of the process participants. Garcia et al. [243] proposed an optimization method for executing business processes on commodity blockchain technology.

Guo and Liang [244] stated that blockchains could revolutionize the payment techniques and credit information systems in banks. Cocco et al. [245] probed the challenges and opportunities of implementing blockchain technology across banking. In their work, they concluded that blockchain could handle financial processes in a more efficient way than the current system, such as Bitcoin. Yermack [246] evaluated the implications of the changes brought by blockchain in corporate governance. The author stated that blockchain could offer benefits including lower cost, greater liquidity, more accurate record-keeping, and transparency of ownership.

Silk Road, the first modern darknet marketplace, can be a typical example of blockchain application in market. In Christin's work [247], he performed a comprehensive measurement analysis of Silk Road. He even showed how items were sold and how seller population evolved over time. Soska and Christin [248] presented a long-term measurement analysis of the online anonymous marketplace ecosystem over more than two years, including 19 different marketplaces. These markets included Silk Road 2.0, Agora, Atlantis, Block Flag, etc.

Government. Blockchain can be applied to government in various sectors [65], [249]. For example, it can bring greater transparency of transactions between government agencies and citizens. After a literature review of Bitcoin researches, Olnes [250] found there was an absence in this field and describes the vision of applying blockchain technology in government management to verify multiple persistent documents in the public sector. Atzori [251] claimed that a comprehensive analysis of blockchain applications about politics was lacking to date. Therefore, the author aimed to fill the gap and discussed the key points of blockchain-based decentralized governance.

Olines et al. [252] offered directions for further research into the potential benefits of blockchain applications in e-government. They discussed the role of governance of blockchain architectures and analyzed how applications could comply with societal needs and public values. Biswas et al. [253] aimed to build a smart city by introducing blockchain. It proposed a security framework that combines blockchain technology with smart devices to provide a secure communication platform for smart cities. Marsal [254] held that blockchain would be the network cities after IoT and smart cities. Lei et al. [255] held that Intelligent Transportation System (ITS) could introduce blockchain technology to the transportation infrastructures with the aim of improving road safety and traffic efficiency. For example, to secure key management in a vehicular communication system, they proposed a framework including a network topology based on a decentralized blockchain structure. Xu et al. [256] propose a blockchain-based resource management framework to save energy consumption in datacenters.

Medical and Charity. Kuo et al. [257] discussed the benefits, pitfalls, and the applications of blockchain in biomedical and healthcare domains. Mettler [258] introduced many starting points of the blockchain technology in the healthcare industry. They illustrated the impact, goals, and objectives of blockchain technology in the field of medical services with examples of public health management, user-oriented medical research, and drug counterfeiting in the pharmaceutical field.

Liu [259] discussed the advantages and disadvantages of blockchain and big data technologies for medical records. Dhillon et al. [260] integrated blockchain into tracking a patient from the first visit to a primary care physician, then to the final diagnosis. Yue et al. [261] proposed a blockchain-based application called a medical data gateway architecture. The system enabled patients to easily and securely own, control and share their own health care data without violating privacy. Azaria et al. [18] proposed MedRec, a new distributed record management system for processing electronic medical records using blockchain technology. The system could provide patients with a comprehensive, immutable log and easy access to medical information from providers and treatment sites. At the same time, it encouraged medical stakeholders (researchers, public health authorities, etc.) to participate in the network as blockchain "miners". Therefore, they could

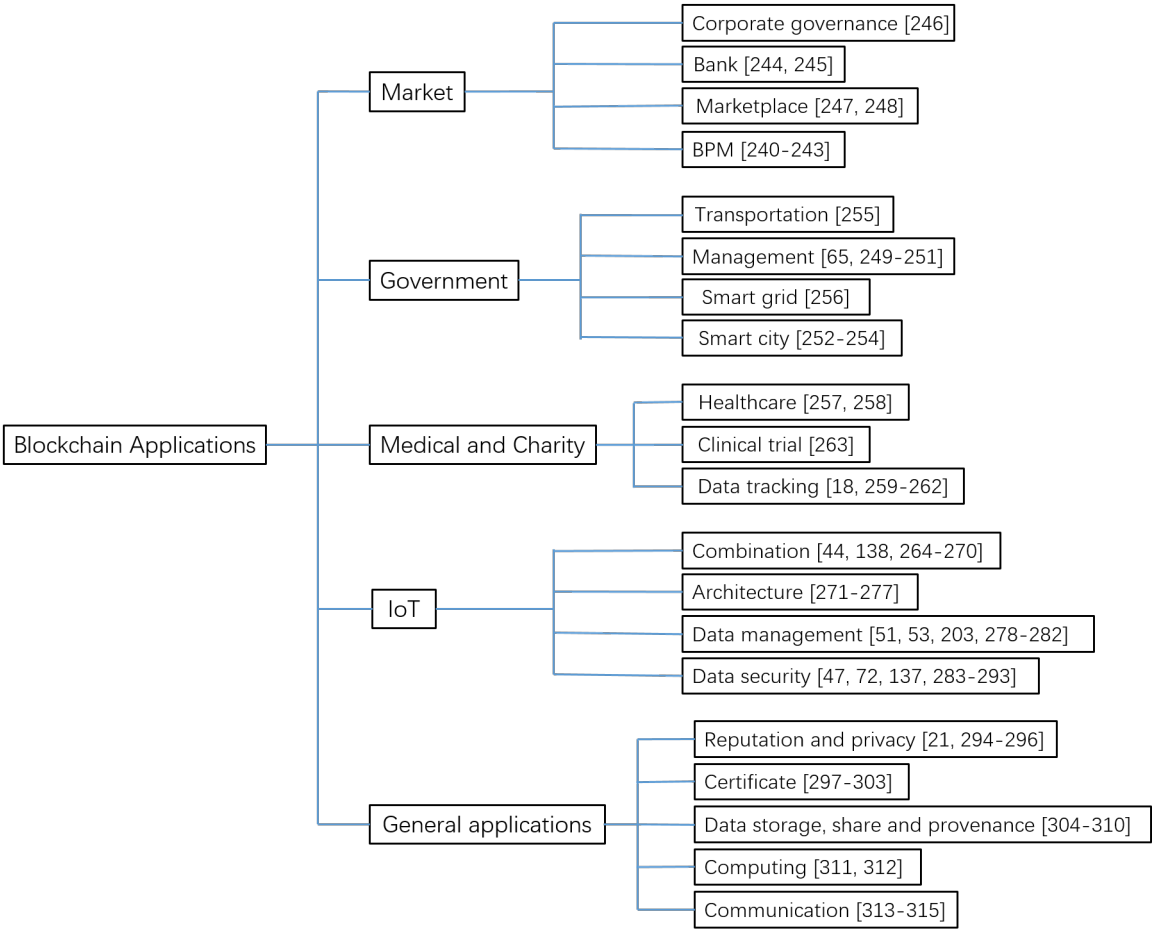


Fig. 19. Blockchain applications in various fields

access aggregated anonymous data as mining rewards as that in PoW to maintain and protect the network. MedRec exploited big data to empower researchers while allowing patients and providers to choose to publish metadata or not. Peterson et al. [262] stated that sharing healthcare data between institutions was challenging because of the compatibility issue brought by the heterogeneous data structure. Therefore, they presented a blockchain-based approach to sharing patient data and predicating consensus on proof of structural and semantic interoperability. Shae et al. [263] presented a blockchain platform architecture for clinical trial and precision medicine. They discussed many aspects of building this platform, including data integrity, identity privacy, distributed and parallel computing, and IoT.

Internet of Things. Christidis and Devetsikiotis [264] held that the blockchain-IoT combination was powerful and could cause significant transformations across several industries, paving the way for new business models and novel distributed applications. In their work, they described that blockchain-IoT combination could facilitate resources sharing, making a marketplace of services between devices become possible. Due to the combination, several existing time-consuming workflow could be automated in a cryptographically verifiable manner. Samaniego et al. [44] held the ability of blockchain to create/store/transfer digital assets in a distributed, decentralized and tamper-proof way was of great practical value for IoT systems. They considered that a key challenge in the deployment of Blockchain as a Service (BaaS) for IoT was the hosting environment. They evaluated the employment of cloud or fog as a platform to handle this challenge. Reyna et al. [265] analyzed the challenges and opportunities when integrating blockchain and IoT. The opportunities brought by blockchain included providing confidence and trust in distributed environments without the third authority. The challenges included storage capacity, scalability, security, and anonymity. Huh et al. [266] utilized blockchain to release the limitations and issues in IoT server-client synchronization. However, regarding the question about whether blockchain can strength IoT, Kshetri [267] gave the answer “maybe”. He explained it in four aspects: IoT security, identity and access management, cloud, and supply chain security. Salimitari and Chatterjee [129] gave an overview of blockchain consensus mechanisms that were applicable to resource-constraint IoT networks and devices. Bahga and Madisetti [268] presented BPIIoT, a decentralized

industrial IoT system based on blockchain technology. Ghuli et al. [269] stated that blockchain technology could be used to transfer the ownership of IoT devices from one owner to another. Ruta et al. [270] exploited blockchain to serve as a semantic resource/service discovery layer to ensure the verifiability of transactions.

Dorri et al. [271], [272], [273] presented a secure, private and lightweight architecture for IoT based on blockchain while eliminating the overhead of blockchain and maintaining the framework security and privacy. Park et al. [274] held blockchain could be applied beyond the IoT environment, and discussed how to adapt blockchain security to cloud computing. Karlsson et al. [275] argued that the Nakamoto-style blockchains were not suitable for application in many IoT scenarios. The drawbacks could be power intensive and high network connectivity in blockchains. Therefore, they suggested a directed acyclic graph (DAG)-structured blockchain called Vegvisir to deal with the above issues. They also stated that Vegvisir gained benefits including partition-tolerant, tamper-proof and data provenance. Montori et al. [276] proposed a collaborative IoT architecture for smart cities and environmental monitoring. Ta-Shma et al. [277] designed a scalable architecture for ingesting and analyzing IoT data, which exploited the historical data to make real-time analysis.

Tosh et al. [203] stated that blockchain's public and distributed peer-to-peer ledger capability could benefit cloud computing services which require functions including data provenance, management of digital assets, auditing and distributed consensus. Samaniego and Deters [278] evaluated using cloud-hosted permission-based blockchains as an approach for persistent data storage and distributing code. Shafagh et al. [279] presented a blockchain-based design for IoT that brought a distributed access control and data management. Zhang and Wen [280] exploited blockchain and smart contract to realize the transaction of smart property and paid data on the IoT. Ozyilmaz et al. [281] proposed a blockchain-based IoT data market to enable the interaction and collaboration of IoT data providers and machine learning users. Danzi et al. [53] held that though blockchain was suitable to support IoT interaction, keeping a local copy was infeasible for power and memory constrained devices. Therefore, they designed lightweight protocols to aggregate blockchain data. Fitzgerald et al. [282] studied data aggregation and dissemination with minimum total energy consumption for IoT. Danzi et al. [51] designed the general architectures and synchronization protocols to enable synchronizing the IoT endpoint data to the blockchain. In their paper, they evaluated the protocol from communication costs and security level.

Dorri et al. [72] held that blockchain has the potential to handle the security and privacy issues in IoT, and they proposed a special tiered lightweight scalable blockchain for it. The similar tiered design can be also found in [47]. Sagirlar et al. [128] pointed out the current centralized cloud controlled IoT platform had many disadvantages, such as security and trust issue, and the maintenance costs. Therefore, they designed Hybrid-IoT, a blockchain architecture for IoT, to overcome these challenges. Pan et al. [283] thought the IoT devices were vulnerable facing malicious hackers. Therefore, they designed "EdgeChain" to exploit permissioned blockchain and smart contracts to overcome the security weaknesses. Pinto et al. [284] implemented a blockchain-based Public Key Infrastructure (PKI) to provide data identity proof to protect the sensitive information in IoT. Manzoor et al. [285] proposed a blockchain-based proxy re-encryption scheme to secure IoT data sharing and tackle the scalability and trust issues. Considering that IoT devices were exposed to attackers, Boudguiga et al. [286] proposed a P2P mechanism to facilitate the loophole patch updates for security reason. Liu et al. [287] pointed out traditional IoT data integrity verification often needed a third party auditor, which was far from satisfactory. Therefore, they exploited blockchain to achieve the same goal without the third party auditor. The similar discuss can also be found in [288]. Pinno et al. [289] held there were many private and confidential data to be protected in IoT. Therefore, they designed a blockchain-based architecture to enable data access control. Novo [290] also proposed a blockchain-based access control system for IoT. Moon et al. [291] developed a random access procedure to enable the massive connectivity of IoT devices. Zhou et al. [292] designed a blockchain-based IoT system named BeeKeeper to secure data storage and computing. Wazid et al. [293] designed a lightweight remote user authentication scheme for hierarchical IoT network.

General applications. We have discussed the major areas for blockchain applications in above sections. In this section, we will introduce specific blockchain applications that can be used in a lot of areas.

- **Reputation and privacy**. Dennis and Owen [294] proposed the first generalized reputation system that could be applied to multiple networks based on blockchain. They also solved many unanswered questions in today's current generation reputation systems. They also claimed that they presented the first generalized reputation system that could be applied to multiple networks based on blockchain. Carboni [295] built a blockchain-based distributed feedback management system to access the reputation of a member in a web community. Zyskind and Nathan [21] utilized blockchain storage to build a personal data management platform for privacy protection. Using blockchain to protect people's privacy is also discussed in [296].
- **Certificate**. Jamthagen and Hell [297] designed the Keyless Signature Infrastructure (KSI), which could provide users with a method to timestamp documents on a per-second basis. The KSI was composed of multiple server-tier global infrastructures, and the root hash could be used to verify the integrity and timestamp of a document. Blockchain could be used as an additional publication layer on top of a KSI to gain faster publication, proof of origin for the publisher. It also enabled the possibility for third parties to explicitly verify root hashes before they were published. Baldi et al. [298] addressed the major failure points of modern PKI by introducing a public, decentralized and robust ledger in which certificate revocation lists were collected. HIBE was used to build name-based security mechanisms for secure distribution of content. All parameters in the system were recorded

using the blockchain as a global transaction list. Ali et al. [299] proposed a blockchain-based naming and storage system design called Blockstack. They stated that Blockstack was an open source software that could power a production PKI system for 55,000 users. In their work, they also described their experience of operating a large deployment of a decentralized PKI service built on top of the Namecoin blockchain. Wilson [300] provided a new bitcoin-based certificate format to solve problems between PGP (Pretty Good Privacy) and related trust networks. The IKP proposed in [301] is based on blockchain-based PKI enhancements that automatically respond to certification authority (CA) misconduct and provide incentives for those who help detect misconduct. To solve the challenges that providing anonymity with identification, authentication, and authorization of the users and protection of their transaction, Muftic [302] introduced BIX (Blockchain Information Exchange) certificates, a type of cryptographically encapsulated objects. Longo et al. [303] analyzed the security of BIX protocol and find some attack scenarios against it.

- **Data store, sharing and provenance**. Blockchain can introduce transparency of copyright ownership chain. It can also mitigate the risks of online privacy by enabling control over digital copy and creating a civilized market for digital content. It can also combine the simplicity of creative commons and open source type of licenses with revenues stream [304]. In [304], Savelyev also argued that the benefit brought by blockchain came at costs. He discussed the legal issues and economic factors that were supposed to be updated. Fotiou and Polyzos [305] presented decentralized name-based security mechanisms that provided content provenance verification and content integrity protection. In their mechanisms, they utilized blockchain to deliver the system parameters. Liu et al. [306] proposed a lightweight blockchain system, LightChain to save resources in industrial IoT scenarios. Tran et al. [307] designed a registry that records a list of information on a blockchain by leveraging key blockchain properties including data integrity, immutability, and availability. Xu et al. [308] believed blockchain was an emerging technology for decentralized and transactional data sharing across a large network of untrusted participants, and they considered blockchain as a software connector to facilitate the data sharing. Pazaitis et al. [309] explored the potential of blockchain technology in enabling a new system of value that would better support the dynamics of social sharing. The similar work can also be found in [310].
- **Computing**. Dong et al. [311] noticed that existing verification computing techniques were expensive to be deployed. To achieve verifiability, at least three clouds were needed to finish a task. Their work proposed a new verification way by leveraging game theory and smart contract, in which only two clouds were needed, thus lowering the verifiability costs. Choudhuri et al. [312] held that in secure multiparty computing that each party provided an output while keeping its own input private in a mutual distrust case, fairness became critical. In their work, they exploited public bulletin boards implemented by blockchain to achieve the fairness goal.
- **Communication**. Rowan et al. [313] proposed side-channel and blockchain public-key infrastructure to improve the security of wireless communication in autonomous and self-driving vehicles. Leiding et al. [314] combined VANET (Vehicular Ad Hoc Network) and Ethereum blockchain-based applications to implement transparent, self-managed and decentralized systems. Xing et al. [315] held that centralized authority abuse presented a risk of deploying origin authentication, which was important for BGP security to resist IP prefix hijacking. Therefore, they proposed BGPCoin, a blockchain based Internet resource management solution to allocate and revoke resources.

VI. Challenges and opportunities

Currently, though the development of blockchain has made great progress, the application of blockchain is far from the ideal scale. The blockchain has been criticized mainly in five aspects: illegal activities, transaction throughput, storage problem, privacy and security. Since blockchain is a fundamental technology that is supposed to revolutionize the future, there will always be opportunities where there are challenges.

A. Illegal activities

Since public blockchain is maintained by a community instead of some authorities, its decentralized nature also makes it an ideal platform for illegal activities under no censor. Blockchain based altcoin economy facilitates money laundering and other illegitimate activities, such as purchasing illicit goods, broadcasting child pornography and Internet gambling. For example, the Silk Road website, an online black market, provides a platform for sellers and buyers to do illegal drug dealing [316]. Meiklejohn et al. [76] discussed a broad class of criminal activities in the thefts and proposed *Heuristic 2* to de-anonymize the transaction flows, thus tracking illicit-obtained bitcoin to exchanges. Cai and Zhu [317] held that blockchain technology could provide new opportunities for redesigning a reputation system and they explored the potential and limitation under various attacks.

B. Transaction throughput

The main factors that affect the transaction throughput are broadcast communication, consensus mechanism and transaction verification. The most important part of these factors is the consensus mechanism. Due to the trust characteristics of blockchain, the guarantee of correctness and uniqueness of blockchain requires each node to prove enough work, so that it can express its reliability and authenticity of its own message. In the process of proof, it is often necessary to consume a lot of time and power, so the speed of processing is limited. Currently, there are many relevant researchers and companies trying to solve this problem. Although the PoW algorithm has solved the problem of double-spending, it wastes too many

resources, and it takes too long to verify the transactions. In above section, we have summarized the works related with blockchain scala, Though many works have been proposed to increase the scalability of blockchain , as we have discussed in above section (i.e., adjusting block size, sharding, off-chain payment), there are still many issues remaining to be solved, such as the security issue accompanying by high throughput.

C. Storage issues

In blockchain, all transactions are recorded in blocks, and the backup of different nodes in the distributed system can effectively prevent the ledger from being tampered with. However, with the accumulation of the transaction and the growth of the data, the remaining space in each block is getting smaller and smaller, and the volume of the block is close to the pre-designed size of 1MB. Weber et al. [318] even identified the availability limitations of Ethereum and Bitcoin, and demonstrated the write availability of both blockchains was low.

When Nakamoto created bitcoin, he had anticipated this ahead of time, but he put his hope on the development of network, storage, and hardware to solve this problem. To show how the data volume will be increased, we can make a simple estimation. Assuming that each transaction is 512 bytes, and the fee unit is 0.0004/KB, then the annual transaction data size will become too expensive to store, as is shown in table IV.

TABLE IV. THE EXPECTED TRANSACTION DATA SIZE WITH THE INCREASE OF TRANSACTION THROUGHPUT

TPS	Unit Block Size	Block Fees	Annual Size
1	0.3MB	0.12 BTC	15 GB
3	0.9MB	0.36 BTC	47 GB
10	3MB	1.2 BTC	150 GB
100	30MB	12 BTC	1.5 TB
1,000	300MB	120 BTC	15 TB
10,000	3GB	1,200 BTC	150 TB
100,000	30GB	12,000 BTC	1,500 TB

According to VISA's record in 2015, a total of 92,064 million payment transactions were generated throughout the year. If converting these network transaction data volume into that in Bitcoin, the annual size could be 47TB, which is far beyond the disk capacity of ordinary machine/database. If increasing the block size as 30 MB, the annual transaction data volume also can be 1.5TB, which is also a huge number. Therefore, it is also a very challenging problem to expand the capacity of the blockchain.

D. Privacy and Security

In public blockchain, every participant can get a complete data backup, and all the transaction data are open and transparent, which guarantees the reliability of the blockchain to a certain extent. However, for many blockchain applications, this feature can be fatal. Usually, users themselves intend to keep their account privacy and transaction information protected. For business organizations, many accounts and transaction information are also important assets and business secrets need to be protected. Both the users and organizations do not intend to publicly share their secrets with their other peers. With the help of big data technology, the encryption technology of blockchain is indeed confirmed that there may exist risks. It has been proved that the anonymization of transaction address still cannot guarantee the anonymity of the users, and some deliberate attacks can still cause threats. Related transactions between different addresses can expose users' private information, which is fatal for commercial use. Moreover, with the development of quantum computing, cryptography itself is also threatened. Therefore, we need to constantly update the technology to ensure the security of blockchain.

A typical concern is the common underlying assumption that the majority of participants are honest. For example, in PoW based blockchains, if an attacker gains more than 50% of the computation power of the whole network, he can revise the transactions in blocks. Then this attacker can easily mount the double-spending and give a devastating strike to these blockchains, of which the Bitcoin will be the first one. It is reported that the total computation power of mining pools in China has been more than 70% [319]. If these mining pools collaboratively mount the double-spending attack, the consequence will be serious. Badertscher et al. [320] also questioned this default majority assumption and provided analyses on the incentives for the miner's behavior. Though blockchains such as Bitcoin take different incentive strategies to stimulate the miners, brute-force attacks (e.g., Goldfinger attack) are still potential risks.

Another issue needs to be considered is the quantum attack. The key cryptographic protocols used to secure the internet and financial transactions today are all susceptible to attack by the development of a sufficiently large quantum computer [321]. For example, Bitcoin security will be broken by the massive computing power of quantum computers within 10 years [322]. Technically, the elliptic curve signature scheme used by Bitcoin could be completely cracked by a quantum computer as early as 2027 [321]. The same risk is also pointed out in [323], [324]. Propov designed *tangle* [325], a directed acyclic graph (DGA) for storing transactions that is resistant to quantum attacks.

E. Combination with IoT

In the above section, we have discussed many applications of blockchain into IoT from various aspects. However, their integration is still a non-trivial job. The above-mentioned challenges (i.e., illegal activities, transaction throughput, storage issues, and privacy and security) can also apply to the integration, and some challenges even become more prominent.

1) Storage issues: The storage issue [265], [278], [292] can be more serious when integrating blockchain with IoT. First, different from normal blockchain nodes (e.g., computers), the majority of IoT devices gain low storage capacity. Through the above analyses of storage issues, even the normal blockchain nodes with high storage capacity cannot easily handle the challenge of continuous increasing data amount. Playing IoT devices with low capacity as blockchain nodes will become more challenging. Second, the heterogeneous IoT devices with various storage capacity make the storage issues

become more complicated. The IoT devices with minimum storage capacity will become the bottleneck. Third, the highly increasing number of IoT connected devices will generate huge amount of data. According to Statista [326], there are 23.14 billion IoT connected devices and this number is projected to amount to 75.44 billion worldwide by 2025. Compared with Bitcoin network that only holds around 10,000 nodes [327], such a huge increasing number further makes the data storage issue become challenging in IoT. Makhdoom et.al. [328] also pointed out blockchain was not designed to store a large amount of data generated by IoT system with thousands of nodes.

2) Transaction throughput: The transaction throughput issue will also become challenging. First, the heterogeneity of IoT devices implies various communication protocols (e.g., Bluetooth, Wi-Fi, 4G), which makes the cost of achieving consensus in blockchain become expensive. To facilitate the communication between different devices, intermediate nodes that can convert one protocol to another is possible to be used. Another approach could be unifying the communication protocol, which is nearly impossible, considering that the existing IoT devices are owned by different bodies. Second, it is non-trivial to synchronize a huge number of IoT devices. The communication cost will become extraordinarily high when each node needs to keep broadcasting its status and data. How to eliminate these obstacles and design good consensus mechanisms to increase the transaction throughput when combining blockchain with IoT are issues remain to be resolved. IOTA [329] is a distributed ledger technology aiming to enable IoT. It adopts *tangle* as its consensus algorithm and can achieve up to 1,000 TPS.

3) Privacy and Security: Since IoT devices are often sensors that detect the environment, there is also a lot of sensitive data being collected. For instance, in a smart home, IoT devices will sense and collect a lot of sensitive data about people's privacy. It is unacceptable when the private data containing people's daily activities are synchronized with the nodes outside the smart home. Another concern is the communication security. Appropriate secure communication protocols are supposed to be designed to thwart eavesdropping and other attacks. The current Transport Layer Security (TLS) or Datagram Transport Layer Security (DTLS) is complex or heavy [265], especially for IoT networks with a large number of devices. One more concern is that IoT devices can be the prey of attackers who maliciously exploit them to launch various attacks. For instance, in 2006, a large number of IoT devices were exploited to DDoS-attack systems operated by Domain Name System (DNS) provider Dyn, which resulted in major Internet platform and services unavailable in Europe and North America [330]. Therefore, in addition to proper secure communication protocols, secure authentication and attack-defense mechanisms are expected to be designed when integrating blockchain with IoT. Kumar and Mallick [331] also believed that blockchain could be utilized to address the security issues in IoT. Khan and Salah [332] held that blockchain could blockchain can eliminate key management and distribution of IoT devices, thus simplifying the security protocols and secure IoT communication. Hammi et al. [333] exploited blockchain to build an authentication system for IoT. Considering that IoT devices could still be compromised even under best security efforts, Banerjee et al. [334] even suggested using blockchain to facilitate IoT self-healing for compromised devices.

F. Combination with hot technology

Blockchain gains great potential to be applied to different fields, thus having the opportunities to be combined with some popular technologies. In this part, the combination of blockchain and hot technologies including quantum computing, edge/fog computing, big data, artificial intelligence are discussed. Through our investigation, we find although there is great potential by combining blockchain with these technologies, there is a limited number of literatures in these research directions.

1) Blockchain and Quantum Computing: Quantum computing is still in its infancy, there will be more risks to blockchain with its development because of its significant computing power. However, there are always chances accompanying risks. A conceptual design for quantum blockchain is proposed in [335]. Ikeda proposed Bitcoin [336], a decentralized online peer-to-peer quantum cash system.

2) Blockchain and Edge/Fog Computing: Edge/Fog Computing is a hot technology aiming to narrow the "distance" between the terminal users with the computing center, which is relatively too far in the traditional cloud model. Stanciu [337] utilized blockchain as a hierarchical and distributed control system for edge computing. Jiao et al. [338] considered deploying edge computing service to support the mobile blockchain. They proposed an auction-based edge computing resource market of the edge computing service provider, where there was competition among miners. Since PoW needed to consume too much energy, Xiong et al. [339] considered edge computing as an enabler for mobile blockchain and exploited a two-stage Stackelberg game to achieve equilibrium of edge computing service provider and individual miners. Sharma et al. [340] proposed a blockchain-based distributed cloud architecture with a software defined networking (SDN) enabling controller fog nodes at the edge of the network to provide availability, real-time data delivery, scalability, security, resilience, and low latency. Sun and Ansari [341] proposed EdgeIoT, an IoT architecture for mobile edge computing to better provide the storage, computing, and communication resources in IoT to users.

3) Blockchain and Big Data: Because of the storage and scalability issues, blockchain based technology is not suitable for big data, where massive and high-velocity data are supposed to be processed. It is hard to imagine that all the blocks need to keep a copy of all the large transaction data, which will result in huge data redundancy. Though the blocks are only expected to store the hash value for large data, it brings new issues for checking the integrity of the data through off-chain operation, which will entail data privacy and other issues. The above-mentioned healthcare data field is a typical example that blockchain joints hands with big data, yet remains a lot of issues to be resolved.

4) Blockchain and AI: Though both AI and blockchain are hot technologies with great potential, there are a limited number of literatures probing their combination, leaving a blank space to be filled in. Combined with AI and big data, blockchain based technology can produce more significant impacts [267]. Swan [342] outlined blockchain thinking as an *input-processing-output* computational system that would benefit AI. For example, deep learning algorithms need to process huge data in the distributed database in some cases (e.g., the medical data), which can be stored and organized by blockchain. Omohundro [343] believed cryptocurrencies and smart contracts might provide an infrastructure for ensuring AI systems to follow specific legal and safety regulations, thus making it become more integrated into human society. Vice versa, AI systems are also needed to transform information from the real world where smart contracts can be depended on. Luong et al. [344] even combined three hot technologies: edge computing, deep learning and blockchain. Specifically, they used deep learning to derive an optimal auction for allocating edge resources to assist the mining process in blockchain.

5) Blockchain and VR: Blockchain can also be combined with Virtual Reality (VR). Since there is a large quantity of data stored in blockchain, these data can be used to facilitate the learning and decision- making processes, especially in systems need continuous participation of users. By utilizing VR, these data can be visualized in a more vivid way, thus enhancing the above processes. Laskowski [345] proposed a blockchain-enabled participatory decision support framework by integrating Agent Based Modeling, blockchain, and VR. In this framework, VR hardware was used to visualize the participatory outcomes. Another combination of blockchain and VR is demonstrated in Decentraland [346], a decentralized virtual reality platform powered by Ethereum blockchain. The virtual assets in this virtual world is stored in blockchain and managed by smart contract.

G. Undergoing blockchain standards

Some international organizations or institutes have been actively devoted to setting blockchain standards to regulate and facilitate its development. IOS/TC 307 [347] is an international technical committee established in 2016, aiming to standardize blockchains and distributed ledger technologies. It is the major blockchain standard developing body, and now has 35 participants and 13 observing members from different countries. In its roadmap, there are 10 standards under development, covering a wide range of blockchain technologies. These standards can be shown in table V. In this table, "Stage" is the international harmonized stage code indicating the stages of standards. For example, the stage of IOS/NP TR 23244 is "10.99", which indicates this new project has been approved.

In addition to the above standards to be developed, there are other standardization attempts by other organizations or institutes around the world.

- *Blockchain–Reference Architecture* [349]. This standard was released by China Blockchain Technology and Industry Development Forum in 2017. It regulates the blockchain reference architecture from a developer's view. Specifically, it defines terms and definitions related with blockchain, and also clarifies the blockchain user interface, function interface, the relationship between them, as well as the roles and function components in each interface. This architecture divides blockchain function into four layers: user layer, service layer, core layer and basic layer. These layers involve blockchain development, maintenance, security, supervision and audit.
- *Blockchain–Data format specification* [350]. This standard was also released by China Blockchain Technology and Industry Development Forum in 2017. It regulates the blockchain data format, including data structure, data taxonomy, data relationship, and data format requirements. Also, it classifies blockchain data into six categories: account data, block data, transaction data, entity data, smart contract data and configuration data.
- *IEEE P2418* [351]. It is a standard for the framework of blockchain use in Internet of Things (IoT). It aims to develop definitions of protocols for blockchain implementations within an IoT architectural framework. The expected framework will include tokens, assets, credential networks, smart contract, etc. The scalability, security and privacy issues are also to be addressed in this project.
- *The Web Ledger Protocol 1.0* [352]. It is published by World Wide Web Consortium (W3C) Blockchain Communication Group to provide format and protocols for decentralized ledgers on the Web. The primary goal of the ledger format is flexibility, thus allowing the pluggability of consensus algorithms, data structures for recording history, and the type of data that can be stored in the ledger.

TABLE V. THE STANDARDS UNDER DEVELOPMENT OF ISO/TC 307

Standard	Target	Stage [348]
ISO/AWI 22739	Terminology	20.00
ISO/NP TR 23244	Overview of privacy and personally identifiable information (PII) protection	10.99
ISO/NP TR 23245	Security risks and vulnerabilities	10.99
ISO/NP TR 23246	Overview of identity management using blockchain and distributed ledger technologies	10.99
ISO/AWI 23257	Reference architecture	20.00
ISO/AWI TS 23258	Taxonomy and Ontology	20.00
ISO/AWI TS 23259	Legally binding smart contracts	20.00
ISO/NP TR 23455	Overview of and interactions between smart contracts in blockchain and distributed ledger technology systems	10.99
ISO/NP TR 23576	Security of digital asset custodians	10.99
ISO/NP TR 23578	Discovery issues related to interoperability	10.99

VII. CONCLUSION

This paper conducts a comprehensive survey of the blockchain works. Specifically, we classify these works into four layers: data layer, consensus layer, network layer, and application layer. In the data layer, we introduce the fundamental elements of blockchain, such as the cryptographic factors, the blockchain data structures, the mining process. In the network layer, we focus on the blockchain P2P network issues and technologies, which include the scalability,

the multi-chain networks, the transaction process and data privacy. We point out that though many works have been done to increase the scalability of blockchain, few of them are practical and secure enough to be deployed in the real system. Meanwhile, the attacks in traditional P2P networks also can be mounted in blockchain's network. Though adopting pseudonym to achieve user anonymity and lots of later works have been done to improve it, it is still a potential threat due to the openness of blockchain. In consensus layer, classical consensus mechanisms (e.g., PoW, PoS, DPoS) are discussed. The advantages and disadvantages of these consensus algorithms are also compared. Meanwhile, since PoW is the original blockchain consensus algorithm and it has been widely deployed in the real blockchain systems, many attacks against it and the countermeasures have been listed in the layer. In the application layer, we introduce the blockchain extensions (e.g., smart contracts, sidechain) and the general blockchain applications. Since blockchain has been applied in various fields, we categorize the related work by the fields. Finally, the challenges and opportunities of blockchain are analyzed to broaden the research scope.

ACKNOWLEDGMENT

This work is supported by National Key Research and Development Program of China (2018YFB1004700); NSFC (61872195, 61872310, 61832005, 61572262, 61872240).

REFERENCES

[1] D. Yaga, P. Mell, N. Roby, and K. Scarfone, "Blockchain technology overview," *Draft NISTIR*, vol. 8202, 2018.

[2] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[3] I. Bentov, R. Pass, and E. Shi, "Snow white: Provably secure proofs of stake." *IACR Cryptology ePrint Archive*, vol. 2016, p. 919, 2016.

[4] D. Larimer, "Delegated proof-of-stake white paper," 2014.

[5] "What is proof of burn (eli5)?" http://slimco.in/proof-of-burn-eli5/.

[6] J. R. Douceur, "The sybil attack," in *International workshop on peer-to-peer systems*. Springer, 2002, pp. 251–260.

[7] P. Otte, M. de Vos, and J. Pouwelse, "Trustchain: A sybil-resistant scalable blockchain," *Future Generation Computer Systems*, 2017.

[8] A. Back, "Hashcash-a denial of service counter-measure," 2002.

[9] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 254–269.

[10] M. Bartoletti and L. Pompianu, "An empirical analysis of smart contracts: platforms, applications, and design patterns," in *International Conference on Financial Cryptography and Data Security*. Springer, 2017, pp. 494–509.

[11] B. Rodrigues, T. Bocek, A. Lareida, D. Hausheer, S. Rafati, and B. Stiller, "A blockchain-based architecture for collaborative ddos mitigation with smart contracts," in *IFIP International Conference on Autonomous Infrastructure, Management and Security*. Springer, 2017, pp. 16–29.

[12] A. Bogner, M. Chanson, and A. Meeuw, "A decentralised sharing app running a smart contract on the ethereum blockchain," in *Proceedings of the 6th International Conference on the Internet of Things*. ACM, 2016, pp. 177–178.

[13] P. McCorry, S. F. Shahandashti, and F. Hao, "A smart contract for boardroom voting with maximum voter privacy," in *International Conference on Financial Cryptography and Data Security*. Springer, 2017, pp. 357–375.

[14] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *Security and Privacy (SP), 2016 IEEE Symposium on*. IEEE, 2016, pp. 839–858.

[15] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timón, and P. Wuille, "Enabling blockchain innovations with pegged sidechains," *URL: http://www. opensciencereview. com/papers/123/enablingblockchain-innovations-with-pegged-sidechains*, 2014.

[16] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer, D. Song, and R. Wattenhofer, "On scaling decentralized blockchains," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 106–125.

[17] Wikipedia, "List of cryptocurrencies," https://en.wikipedia.org/wiki/List_of_cryptocurrencies, 2018.

[18] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "Medrec: Using blockchain for medical data access and permission management," in *Open and Big Data (OBD), International Conference on*. IEEE, 2016, pp. 25–30.

[19] S. A. Abeyratne and R. P. Monfared, "Blockchain ready manufacturing supply chain using distributed ledger," 2016.

[20] J. Chen, S. Yao, Q. Yuan, K. He, S. Ji, and R. Du, "Certchain: Public and efficient certificate audit based on blockchain for tls connections."

[21] G. Zyskind and O. Nathan, "Decentralizing privacy: Using blockchain to protect personal data," in *Security and Privacy Workshops (SPW), 2015 IEEE*. IEEE, 2015, pp. 180–184.

[22] E. Commission, https://blogs.ec.europa.eu/eupolicylab/portfolios/blockchain4eu/.

[23] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2084–2123, 2016.

[24] M. Holub and J. Johnson, "Bitcoin research across disciplines," *The Information Society*, vol. 34, no. 2, pp. 114–126, 2018.

[25] A. Baliga, "Understanding blockchain consensus models," Tech. rep., Persistent Systems Ltd, Tech. Rep, Tech. Rep., 2017.

[26] N. Atzei, M. Bartoletti, and T. Cimoli, "A survey of attacks on ethereum smart contracts (sok)," in *International Conference on Principles of Security and Trust*. Springer, 2017, pp. 164–186.

[27] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, "Untangling blockchain: A data processing view of blockchain systems," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 7, pp. 1366–1385, 2018.

[28] L. Pacioli, *Summa de Arithmetica geometria proportioni: et proportionalita...* Paganino de Paganini, 1994.

[29] Wikipedia, "Blockchain," https://en.wikipedia.org/wiki/Blockchain#cite_note-te20151031-1.

[30] "How anonymous are bitcoin transactions?" https://bitcoin.stackexchange.com/questions/52/how-anonymous-are-bitcoin-transactions.

[31] E. Androulaki, G. O. Karame, M. Roeschlin, T. Scherer, and S. Capkun, "Evaluating user privacy in bitcoin," in *International Conference on Financial Cryptography and Data Security*. Springer, 2013, pp. 34–51.

[32] M. Ober, S. Katzenbeisser, and K. Hamacher, "Structure and anonymity of the bitcoin transaction graph," *Future internet*, vol. 5, no. 2, pp. 237–250, 2013.

[33] M. tree, https://en.wikipedia.org/wiki/Merkle_tree.

[34] D. Schwartz, N. Youngs, and A. Britto, "The ripple protocol consensus algorithm," *Ripple Labs Inc White Paper*, vol. 5, 2014.

[35] M. Rosenfeld, "Analysis of bitcoin pooled mining reward systems," *arXiv preprint arXiv:1112.4980*, 2011.

[36] F. (blockchain), https://en.wikipedia.org/wiki/Fork_(blockchain)#Hard_fork.

[37] F. Coppola, "A painful lesson for the ethereum community," https://www.forbes.com/sites/francescoppola/2016/07/21/a-painful-lesson-for-the-ethereum-community/#5445b9dabb24, 2016.

[38] J. Lanz, "Ethereum community considers hard fork to combat bitmain's asic miner," https://cryptoslate.com/ethereum-hard-form-bitmain-asic-miner/, 2018.

[39] P. Wuille, "Strict der signatures," https://github.com/bitcoin/bips/blob/master/bip-0066.mediawiki, 2015.

[40] M. Carlsten, H. Kalodner, S. M. Weinberg, and A. Narayanan, "On the instability of bitcoin without the block reward," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 154–167.

[41] R. Pass and A. Shelat, "Micropayments for decentralized currencies," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-16, 2015*, 2015, pp. 207–218.

[42] U. Author, "Base58check encoding," 2014.

[43] N. Szabo, "Smart contracts: building blocks for digital markets," *EXTROPY: The Journal of Transhumanist Thought,(16)*, 1996.

[44] M. Samaniego and R. Deters, "Blockchain as a service for iot," in *Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2016 IEEE International Conference on*. IEEE, 2016, pp. 433–436.

[45] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," *draft version 0.5*, vol. 9, p. 14, 2016.

[46] K. Wang, Y. Wang, Y. Sun, S. Guo, and J. Wu, "Green industrial internet of things architecture: An energy-efficient perspective," *IEEE Communications Magazine*, vol. 54, no. 12-Supp, pp. 48–54, 2016.

[47] Z. Bao, W. Shi, D. He, and K. R. Choo, "Iotchain: A three-tier blockchain-based iot security architecture," *CoRR*, vol. abs/1806.02008, 2018. [Online]. Available: http://arxiv.org/abs/1806.02008

[48] V. Daza, R. D. Pietro, I. Klimek, and M. Signorini, "CONNECT: contextual name discovery for blockchain-based services in the iot," in *IEEE International Conference on Communications, ICC 2017, Paris, France, May 21-25, 2017*, pp. 1–6.

[49] T. B. community, "Stratum mining protocol," https://en.bitcoin.it/wiki/Stratum, 2014.

[50] A. Gervais, G. Karame, S. Capkun, and V. Capkun, "Is bitcoin a decentralized currency?" *IEEE security & privacy*, vol. 12, no. 3, pp. 54–60, 2014.

[51] P. Danzi, A. E. Kalør, C. Stefanovic, and P. Popovski, "Analysis of the communication traffic for blockchain synchronization of iot devices," in *2018 IEEE International Conference on Communications, ICC 2018, Kansas City, MO, USA, May 20-24, 2018*, 2018, pp. 1–7.

[52] P. Danzi, A. E. Kalor, C. Stefanovic, and P. Popovski, "Analysis of the communication traffic for blockchain synchronization of iot devices," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–7.

[53] P. Danzi, A. E. Kalør, C. Stefanovic, and P. Popovski, "Delay and communication tradeoffs for blockchain systems with lightweight iot clients," *CoRR*, vol. abs/1807.07422, 2018.

[54] P. K. Sharma, S. Singh, Y. Jeong, and J. H. Park, "Distblocknet: A distributed blockchains-based secure SDN architecture for iot networks," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 78–85.

[55] X. Liang, J. Zhao, S. Shetty, and D. Li, "Towards data assurance and resilience in iot using blockchain," in *MILCOM 2017-2017 IEEE Military Communications Conference (MILCOM)*. IEEE, 2017, pp. 261–266.

[56] K. Kataoka, S. Gangwar, and P. Podili, "Trust list: Internet-wide and distributed iot traffic management using blockchain and SDN," in *4th IEEE World Forum on Internet of Things, WF-IoT 2018, Singapore, February 5-8, 2018*, pp. 296–301.

[57] M. Singh and S. Kim, "Trust bit: Reward-based intelligent vehicle commination using blockchain paper," in *4th IEEE World Forum on Internet of Things, WF-IoT 2018, Singapore, February 5-8, 2018*, 2018, pp. 62–67.

[58] J. A. D. Donet, C. Pérez-Sola, and J. Herrera-Joancomartí, "The bitcoin p2p network," in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 87–102.

[59] C. Ting, Z. Yuxiao, L. Zihao, C. Jiachi, L. Xiaoqi, L. Xiapu, L. Xiaodong, and X. Zhang, "Understanding ethereum via graph analysis," in *INFOCOM 2018-IEEE Conference on Computer Communications, IEEE*. IEEE, 2018.

[60] B. Huang, Z. Liu, J. Chen, A. Liu, Q. Liu, and Q. He, "Behavior pattern clustering in blockchain networks," *Multimedia Tools and Applications*, vol. 76, no. 19, pp. 20 099–20 110, 2017.

[61] S. Feld, M. Schönfeld, and M. Werner, "Analyzing the deployment of bitcoin's p2p network under an as-level perspective," *Procedia Computer Science*, vol. 32, pp. 1121–1126, 2014.

[62] K. Alabi, "Digital blockchain networks appear to be following metcalfe's law," *Electronic Commerce Research and Applications*, vol. 24, pp. 23–29, 2017.

[63] BlockStream, "What we build," https://blockstream.com/technology/.

[64] H. Jin, X. Dai, and J. Xiao, "Towards a novel architecture for enabling interoperability amongst multiple blockchains," in *38th IEEE International Conference on Distributed Computing Systems, ICDCS 2018, Vienna, Austria, July 2-6, 2018*, 2018, pp. 1203–1211.

[65] U. G. C. S. Adviser, "Distributed ledger technology, beyond block chain," https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/492972/gs-16-1-distributed-ledger-technology.pdf, 2015.

[66] H. Sun, H. Mao, X. Bai, Z. Chen, K. Hu, and W. Yu, "Multi-blockchain model for central bank digital currency," in *Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2017 18th International Conference on*. IEEE, 2017, pp. 360–367.

[67] MelonPort, https://melonport.com/.

[68] "anion," https://aion.network/.

[69] Z. Techonology, https://www.zhongan.io/en/.

[70] InterLedger, https://interledger.org/.

[71] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "Sok: Research perspectives and challenges for bitcoin and cryptocurrencies," in *Security and Privacy (SP), 2015 IEEE Symposium on*. IEEE, 2015, pp. 104–121.

[72] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "LSB: A lightweight scalable blockchain for iot security and privacy," *CoRR*, vol. abs/1712.02969, 2017.

[73] T. Golomb, Y. Mirsky, and Y. Elovici, "Ciota: Collaborative iot anomaly detection via blockchain," *CoRR*, vol. abs/1803.03807, 2018.

[74] J. Herrera-Joancomartí, "Research and challenges on bitcoin anonymity," in *Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance*. Springer, 2015, pp. 3–16.

[75] M. C. K. Khalilov and A. Levi, "A survey on anonymity and privacy in bitcoin-like digital cash systems," *IEEE Communications Surveys & Tutorials*, 2018.

[76] S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G. M. Voelker, and S. Savage, "A fistful of bitcoins: characterizing payments among men with no names," in *Proceedings of the 2013 conference on Internet measurement conference*. ACM, 2013, pp. 127–140.

[77] D. Ron and A. Shamir, "How did dread pirate roberts acquire and protect his bitcoin wealth?" in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 3–15.

[78] M. A. Harlev, H. Sun Yin, K. C. Langenheldt, R. Mukkamala, and R. Vatrapu, "Breaking bad: De-anonymising entity types on the bitcoin blockchain using supervised machine learning," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[79] D. Ron and A. Shamir, "Quantitative analysis of the full bitcoin transaction graph," in *International Conference on Financial Cryptography and Data Security*. Springer, 2013, pp. 6–24.

[80] M. Fleder, M. S. Kester, and S. Pillai, "Bitcoin transaction graph analysis," *arXiv preprint arXiv:1502.01657*, 2015.

[81] M. Spagnuolo, F. Maggi, and S. Zanero, "Bitiodine: Extracting intelligence from the bitcoin network," in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 457–468.

[82] F. Reid and M. Harrigan, "An analysis of anonymity in the bitcoin system," in *Security and privacy in social networks*. Springer, 2013, pp. 197–223.

[83] P. Koshy, D. Koshy, and P. McDaniel, "An analysis of anonymity in bitcoin using p2p network traffic," in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 469–485.

[84] A. Biryukov, D. Khovratovich, and I. Pustogarov, "Deanonymisation of clients in bitcoin p2p network," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 15–29.

[85] S. Goldfeder, H. Kalodner, D. Reisman, and A. Narayanan, "When the cookie meets the blockchain: Privacy risks of web payments via cryptocurrencies," *Proceedings on Privacy Enhancing Technologies*, vol. 2018, no. 4, pp. 179–199, 2018.

[86] M. Moser, R. Bohme, and D. Breuker, "An inquiry into money laundering tools in the bitcoin ecosystem," in *eCrime Researchers Summit (eCRS), 2013*. IEEE, 2013, pp. 1–14.

[87] C. Xu, K. Wang, P. Li, S. Guo, J. Luo, B. Ye, and M. Guo, "Making big data open in edges: A resource-efficient blockchain-based approach," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 4, pp. 870–882, 2019.

[88] "The bitcoin laundry," http://www.bitcoinlaundry.com/.

[89] E. Heilman, F. Baldimtsi, and S. Goldberg, "Blindly signed contracts: Anonymous on-blockchain and off-blockchain bitcoin transactions," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 43–60.

[90] E. Heilman, L. Alshenibr, F. Baldimtsi, A. Scafuro, and S. Goldberg, "Tumblebit: An untrusted bitcoin-compatible anonymous payment hub," in *Network and Distributed System Security Symposium*, 2017.

[91] J. Camenisch, M. Drijvers, and M. Dubovitskaya, "Practical uc-secure delegatable credentials with attributes and their application to blockchain," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 683–699.

[92] H. Li, K. Wang, T. Miyazaki, C. Xu, S. Guo, and Y. Sun, "Trust-enhanced content delivery in blockchain-based information-centric networking," *IEEE Network*, 2019.

[93] bitlaundry, http://bitlaundry.com/.

[94] J. Bonneau, A. Narayanan, A. Miller, J. Clark, J. A. Kroll, and E. W. Felten, "Mixcoin: Anonymity for bitcoin with accountable mixes," in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 486–504.

[95] M. Green and I. Miers, "Bolt: Anonymous payment channels for decentralized currencies," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 473–489.

[96] T. Ruffing, P. Moreno-Sanchez, and A. Kate, "P2p mixing and unlinkable bitcoin transactions." *IACR Cryptology ePrint Archive*, vol. 2016, p. 824, 2016.

[97] G. Bissias, A. P. Ozisik, B. N. Levine, and M. Liberatore, "Sybil-resistant mixing for bitcoin," in *Proceedings of the 13th Workshop on Privacy in the Electronic Society*. ACM, 2014, pp. 149–158.

[98] E. Cecchetti, F. Zhang, Y. Ji, A. Kosba, A. Juels, and E. Shi, "Solidus: Confidential distributed ledger transactions via pvorm," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 701–717.

[99] I. Miers, C. Garman, M. Green, and A. D. Rubin, "Zerocoin: Anonymous distributed e-cash from bitcoin," in *Security and Privacy (SP), 2013 IEEE Symposium on*. IEEE, 2013, pp. 397–411.

[100] E. B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *Security and Privacy (SP), 2014 IEEE Symposium on*. IEEE, 2014, pp. 459–474.

[101] E. Fujisaki and K. Suzuki, "Traceable ring signature," in *International Workshop on Public Key Cryptography*. Springer, 2007, pp. 181–200.

[102] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret: Theory and applications of ring signatures," in *Theoretical Computer Science*. Springer, 2006, pp. 164–186.

[103] T. Okamoto and K. Ohta, "Universal electronic cash," in *Annual International Cryptology Conference*. Springer, 1991, pp. 324–337.

[104] N. v. Saberhagen, "Crypto note v 2.0," *CryptoNote. org.[Online]*, vol. 17, no. 10, 2013.

[105] A. Kumar, C. Fischer, S. Tople, and P. Saxena, "A traceability analysis of monero's blockchain," in *European Symposium on Research in Computer Security*. Springer, 2017, pp. 153–173.

[106] S. Noether and A. Mackenzie, "A note on chain reactions in traceability in cryptonote 2.0," *Research Bulletin MRL-0001. Monero Research Lab*, 2014.

[107] J. Macheta, S. Noether, S. Noether, and J. Smooth, "Counterfeiting via merkle tree exploits within virtual currencies employing the cryptonote protocol," 2014.

[108] S. Noether and S. Noether, "Monero is not that mysterious," *Technical report*, 2014.

[109] A. Mackenzie, S. Noether, and M. C. Team, "Improving obfuscation in the cryptonote protocol," *Monero Research Lab, Tech. Rep*, 2015.

[110] S. Noether and A. Mackenzie, "Ring confidential transactions," *Ledger*, vol. 1, pp. 1–18, 2016.

[111] J. K. Liu, V. K. Wei, and D. S. Wong, "Linkable spontaneous anonymous group signature for ad hoc groups," in *Australasian Conference on Information Security and Privacy*. Springer, 2004, pp. 325–335.

[112] S.-N. MRL, "Ring ct for monero."

[113] A. Bancroft and P. Scott Reid, "Challenging the techno-politics of anonymity: the case of cryptomarket users," *Information, Communication & Society*, vol. 20, no. 4, pp. 497–512, 2017.

[114] R. Matzutt, O. Hohlfeld, M. Henze, R. Rawiel, J. H. Ziegeldorf, and K. Wehrle, "Poster: I don't want that content! on the risks of exploiting bitcoin's blockchain as a content store," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. ACM, 2016, pp. 1769–1771.

[115] J. J. Xu, "Are blockchains immune to all malicious attacks?" *Financial Innovation*, vol. 2, no. 1, p. 25, 2016.

[116] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," *Ad hoc networks*, vol. 1, no. 2-3, pp. 293–315, 2003.

[117] A. Singh, T. Ngan, P. Druschel, and D. S. Wallach, "Eclipse attacks on overlay networks: Threats and defenses," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 23-29 April 2006, Barcelona, Catalunya, Spain*, 2006.

[118] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on bitcoin's peer-to-peer network." in *USENIX Security Symposium*, 2015, pp. 129–144.

[119] K. Nayak, S. Kumar, A. Miller, and E. Shi, "Stubborn mining: Generalizing selfish mining and combining with an eclipse attack," in *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*. IEEE, 2016, pp. 305–320.

[120] D. Plohmann and E. Gerhards-Padilla, "Case study of the miner botnet," in *Cyber Conflict (CYCON), 2012 4th International Conference on*. IEEE, 2012, pp. 1–16.

[121] M. Vasek, M. Thornton, and T. Moore, "Empirical analysis of denial-of-service attacks in the bitcoin ecosystem," in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 57–71.

[122] B. Johnson, A. Laszka, J. Grossklags, M. Vasek, and T. Moore, "Game-theoretic analysis of ddos attacks against bitcoin mining pools," in *International Conference on Financial Cryptography and Data Security*. Springer, 2014, pp. 72–86.

[123] A. Laszka, B. Johnson, and J. Grossklags, "When bitcoin mining pools run dry," in *International Conference on Financial Cryptography and Data Security*. Springer, 2015, pp. 63–77.

[124] K. Wang, M. Du, Y. Sun, A. Vinel, and Y. Zhang, "Attack detection and distributed forensics in machine-to-machine networks," *IEEE Network*, vol. 30, no. 6, pp. 49–55, 2016.

[125] B. Lee and J.-H. Lee, "Blockchain-based secure firmware update for embedded devices in an internet of things environment," *The Journal of Supercomputing*, vol. 73, no. 3, pp. 1152–1167, 2017.

[126] Bitpay, "Passwordless authentication using bitcoin cryptography," https://github.com/bitpay/bitauth, 2014.

[127] V. L. Lemieux, "Trusting records: is blockchain technology the answer?" *Records Management Journal*, vol. 26, no. 2, pp. 110–139, 2016.

[128] G. Sagirlar, B. Carminati, E. Ferrari, J. D. Sheehan, and E. Ragnoli, "Hybrid-iot: Hybrid blockchain architecture for internet of things - pow sub-blockchains," *CoRR*, vol. abs/1804.03903, 2018.

[129] M. Salimitari and M. Chatterjee, "An overview of blockchain and consensus protocols for iot networks," *CoRR*, vol. abs/1809.05613, 2018.

[130] V. Gramoli, "From blockchain consensus back to byzantine consensus," *Future Generation Computer Systems*, 2017.

[131] W. Wang, D. T. Hoang, Z. Xiong, D. Niyato, P. Wang, P. Hu, and Y. Wen, "A survey on consensus mechanisms and mining management in blockchain networks," *CoRR*, vol. abs/1805.02707, 2018.

[132] L. Lamport, "How to make a multiprocessor computer that correctly executes multiprocess progranm," *IEEE transactions on computers*, no. 9, pp. 690–691, 1979.

[133] M. P. Herlihy and J. M. Wing, "Linearizability: A correctness condition for concurrent objects," *ACM Transactions on Programming Languages and Systems (TOPLAS)*, vol. 12, no. 3, pp. 463–492, 1990.

[134] M. J. Fischer, N. A. Lynch, and M. S. Paterson, "Impossibility of distributed consensus with one faulty process," *Journal of the ACM (JACM)*, vol. 32, no. 2, pp. 374–382, 1985.

[135] S. Gilbert and N. Lynch, "Perspectives on the cap theorem," *Computer*, vol. 45, no. 2, pp. 30–36, 2012.

[136] L. Lamport, "The part-time parliament," *ACM Transactions on Computer Systems (TOCS)*, vol. 16, no. 2, pp. 133–169, 1998.

[137] D. Ongaro and J. K. Ousterhout, "In search of an understandable consensus algorithm." in *USENIX Annual Technical Conference*, 2014, pp. 305–319.

[138] M. Castro, B. Liskov *et al.*, "Practical byzantine fault tolerance," in *OSDI*, vol. 99, 1999, pp. 173–186.

[139] G. Bissas, B. Levine, A. P. Ozisik, G. Andresen, and A. Houmansadr, "An analysis of attacks on blockchain consensus (draft)," *arXiv preprint arXiv:1610.07985*, 2016.

[140] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 3–16.

[141] M. Rosenfeld, "Analysis of hashrate-based double spending," *arXiv preprint arXiv:1402.2009*, 2014.

[142] R. Pass, L. Seeman, and A. Shelat, "Analysis of the blockchain protocol in asynchronous networks," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2017, pp. 643–673.

[143] J. Garay, A. Kiayias, and N. Leonardos, "The bitcoin backbone protocol: Analysis and applications," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2015, pp. 281–310.

[144] Y. Sompolinsky and A. Zohar, "Secure high-rate transaction processing in bitcoin," in *International Conference on Financial Cryptography and Data Security*. Springer, 2015, pp. 507–527.

[145] Y. Sompolinsky and A. Zohar, "Bitcoin's security model revisited," *arXiv preprint arXiv:1605.09193*, 2016.

[146] A. Kiayias and G. Panagiotakos, "On trees, chains and fast transactions in the blockchain." *IACR Cryptology ePrint Archive*, vol. 2016, p. 545, 2016.

[147] A. Kiayias, E. Koutsoupias, M. Kyropoulou, and Y. Tselekounis, "Blockchain mining games," in *Proceedings of the 2016 ACM Conference on Economics and Computation*. ACM, 2016, pp. 365–382.

[148] J. A. Kroll, I. C. Davey, and E. W. Felten, "The economics of bitcoin mining, or bitcoin in the presence of adversaries," in *Proceedings of WEIS*, vol. 2013, 2013, p. 11.

[149] Y. Lewenberg, Y. Bachrach, Y. Sompolinsky, A. Zohar, and J. S. Rosenschein, "Bitcoin mining pools: A cooperative game theoretic analysis," in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2015, pp. 919–927.

[150] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. bft replication," in *International Workshop on Open Problems in Network Security*. Springer, 2015, pp. 112–125.

[151] M. Chase and S. Meiklejohn, "Transparency overlays and applications," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 168–179.

[152] A. Kiayias, A. Russell, B. David, and R. Oliynykov, "Ouroboros: A provably secure proof-of-stake blockchain protocol," in *Annual International Cryptology Conference*. Springer, 2017, pp. 357–388.

[153] B. David, P. Gaži, A. Kiayias, and A. Russell, "Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2018, pp. 66–98.

[154] S. Barber, X. Boyen, E. Shi, and E. Uzun, "Bitter to better—how to make bitcoin a better currency," in *International Conference on Financial Cryptography and Data Security*. Springer, 2012, pp. 399–414.

[155] QuantumMechanic, "Proof of stake," https://bitcointalk.org/index.php?topic=27787.0.x.

[156] Vandroiy, "mining tragedy," https://bitcointalk.org/index.php?topic=6284.

[157] novacoin, http://novacoin.org/.

[158] yacoin, https://yacoin.org/.

[159] cosmos, https://cosmos.network/.

[160] Nxt, https://nxtplatform.org/.

[161] P. Vasin, "Blackcoin's proof-of-stake protocol v2," *URL: https://blackcoin. co/blackcoin-pos-protocol-v2-whitepaper.pdf*, 2014.

[162] dantheman, "Dpos consensus algorithm - the missing white paper," https://steemit.com/dpos/@dantheman/dpos-consensus-algorithm-this-missing-white-paper.

[163] Bitcoin, "Max wright's thoughts on the security threats of delegated proof of stake and bitshares," http://successcouncil.com/post.php?info=Max-Wrights-thoughts-on-the-Security-threats-of-Delegated-Proof-of-Stake-and-Bitshares.

[164] M. Milutinovic, W. He, H. Wu, and M. Kanwal, "Proof of luck: an efficient blockchain consensus protocol," in *Proceedings of the 1st Workshop on System Software for Trusted Execution*. ACM, 2016, p. 2.

[165] S. Dziembowski, S. Faust, V. Kolmogorov, and K. Pietrzak, "Proofs

of space," in *Annual Cryptology Conference*. Springer, 2015, pp. 585–605.

[166] S. Park, K. Pietrzak, A. Kwon, J. Alwen, G. Fuchsbauer, and P. Gazi, "Spacemint: A cryptocurrency based on proofs of space," *IACR Cryptology ePrint Archive*, vol. 2015, p. 528, 2015.

[167] "chia," https://chia.net/.

[168] I. Bentov, C. Lee, A. Mizrahi, and M. Rosenfeld, "Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract] y," *ACM SIGMETRICS Performance Evaluation Review*, vol. 42, no. 3, pp. 34–37, 2014.

[169] A. Chepurnoy, T. Duong, L. Fan, and H.-S. Zhou, "Twinscoin: A cryptocurrency via proof-of-work and proof-of-stake." *IACR Cryptology ePrint Archive*, vol. 2017, p. 232, 2017.

[170] T. Duong, L. Fan, and H.-S. Zhou, "2-hop blockchain: Combining proof-of-work and proof-of-stake securely," Cryptology ePrint Archive, Report 2016/716, 2016. https://eprint. iacr. org/2016/716, Tech. Rep., 2016.

[171] "Nem technical reference," https://nem.io/wp-content/themes/nem/files/NEM_techRef.pdf.

[172] A. N. Nikolakopoulos and J. D. Garofalakis, "Ncdawarerank: a novel ranking method that exploits the decomposable structure of the web," in *Proceedings of the sixth ACM international conference on Web search and data mining*. ACM, 2013, pp. 143–152.

[173] "Poet 1.0 specification," https://sawtooth.hyperledger.org/docs/core/releases/1.0/architecture/poet.html.

[174] H. Sukhwani, J. M. Martínez, X. Chang, K. S. Trivedi, and A. Rindos, "Performance modeling of pbft consensus process for permissioned blockchain network (hyperledger fabric)," in *Reliable Distributed Systems (SRDS), 2017 IEEE 36th Symposium on*. IEEE, 2017, pp. 253–255.

[175] A. Miller, Y. Xia, K. Croman, E. Shi, and D. Song, "The honey badger of bft protocols," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 31–42.

[176] "Tendermint consensus overview." [Online]. Available: http://tendermint.readthedocs.io/projects/tools/en/master/introduction.html\#consensus-overview

[177] J. Kwon, "Tendermint: Consensus without mining," *Retrieved May*, vol. 18, p. 2017, 2014.

[178] P. Todd, "Ripple protocol consensus algorithm review," *Ripple Labs Inc White Paper (May, 2015) https://raw.githubusercontent.com/petertodd/ripple-consensus-analysis-paper/master/paper. pdf*, 2015.

[179] F. Armknecht, G. O. Karame, A. Mandal, F. Youssef, and E. Zenner, "Ripple: Overview and outlook," in *International Conference on Trust and Trustworthy Computing*. Springer, 2015, pp. 163–180.

[180] D. Mazieres, "The stellar consensus protocol: A federated model for internet-level consensus," *Stellar Development Foundation*, 2015.

[181] "Yobichain," https://github.com/Primechain/yobichain.

[182] J. Herrera-Joancomartí and C. Pérez-Solà, "Privacy in bitcoin transactions: new challenges from blockchain scalability solutions," in *Modeling Decisions for Artificial Intelligence*. Springer, 2016, pp. 26–44.

[183] G. O. Karame, E. Androulaki, and S. Capkun, "Double-spending fast payments in bitcoin," in *Proceedings of the 2012 ACM conference on Computer and communications security*. ACM, 2012, pp. 906–917.

[184] G. O. Karame, E. Androulaki, M. Roeschlin, A. Gervais, and S. Čapkun, "Misbehavior in bitcoin: A study of double-spending and accountability," *ACM Transactions on Information and System Security (TISSEC)*, vol. 18, no. 1, p. 2, 2015.

[185] G. Malavolta, P. Moreno-Sanchez, A. Kate, M. Maffei, and S. Ravi, "Concurrency and privacy with payment-channel networks," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 455–471.

[186] P. Forte, D. Romano, and G. Schmid, "Beyond bitcoin-part ii: Blockchain-based systems without mining." *IACR Cryptology ePrint Archive*, vol. 2016, p. 747, 2016.

[187] "Bips," https://github.com/bitcoin/bips.

[188] M. Rouse, "sharding," https://searchcloudcomputing.techtarget.com/definition/sharding.

[189] Aeternity, "Aternity blockchain," https://aeternity.com/aeternity-blockchain-whitepaper.pdf.

[190] K. Chodorow, *Scaling MongoDB: Sharding, Cluster Setup, and Administration*. " O'Reilly Media, Inc.", 2011.

[191] T. McConaghy, R. Marques, A. Müller, D. De Jonghe, T. McConaghy, G. McMullen, R. Henderson, S. Bellemare, and A. Granzotto, "Bigchaindb: a scalable blockchain database," *white paper, BigChainDB*, 2016.

[192] L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, and P. Saxena, "A secure sharding protocol for open blockchains," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 17–30.

[193] A. E. Gencer, R. van Renesse, and E. G. Sirer, "Service-oriented sharding with aspen," *arXiv preprint arXiv:1611.06816*, 2016.

[194] C. Decker and R. Wattenhofer, "A fast and scalable payment network with bitcoin duplex micropayment channels," in *Stabilization, Safety, and Security of Distributed Systems - 17th International Symposium, SSS 2015, Edmonton, AB, Canada, August 18-21, 2015, Proceedings*, 2015, pp. 3–18.

[195] A. Gervais, H. Ritzdorf, G. O. Karame, and S. Capkun, "Tampering with the delivery of blocks and transactions in bitcoin," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 692–705.

[196] T. Lajoie-Mazenc, R. Ludinard, and E. Anceaume, "Handling bitcoin conflicts through a glimpse of structure," in *Proceedings of the Symposium on Applied Computing*. ACM, 2017, pp. 444–449.

[197] C. Pinzón and C. Rocha, "Double-spend attack models with time advantange for bitcoin," *Electronic Notes in Theoretical Computer Science*, vol. 329, pp. 79–103, 2016.

[198] N. T. Courtois, "On the longest chain rule and programmed self-destruction of crypto currencies," *arXiv preprint arXiv:1405.0534*, 2014.

[199] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," in *International conference on financial cryptography and data security*. Springer, 2014, pp. 436–454.

[200] A. Sapirshtein, Y. Sompolinsky, and A. Zohar, "Optimal selfish mining strategies in bitcoin," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 515–532.

[201] J. Göbel, H. P. Keeler, A. E. Krzesinski, and P. G. Taylor, "Bitcoin blockchain dynamics: The selfish-mine strategy in the presence of propagation delay," *Performance Evaluation*, vol. 104, pp. 23–41, 2016.

[202] N. T. Courtois and L. Bahack, "On subversive miner strategies and block withholding attack in bitcoin digital currency," *arXiv preprint arXiv:1402.1718*, 2014.

[203] D. K. Tosh, S. Shetty, X. Liang, C. A. Kamhoua, K. A. Kwiat, and L. Njilla, "Security implications of blockchain cloud with analysis of block withholding attack," in *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. IEEE Press, 2017, pp. 458–467.

[204] Y. Kwon, D. Kim, Y. Son, E. Vasserman, and Y. Kim, "Be selfish and avoid dilemmas: Fork after withholding (faw) attacks on bitcoin," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 195–209.

[205] H. Finney, "The finney attack," https://bitcointalk.org/index.php?topic=3441.msg48384\#msg48384., 2011.

[206] vector76, "The vector76 attack," https://bitcointalk.org/index.php?topic=36788.msg463391\#msg463391, 2011.

[207] D. Bradbury, "The problem with bitcoin," *Computer Fraud & Security*, vol. 2013, no. 11, pp. 5–8, 2013.

[208] A. Miller, "Feather-forks: enforcing a blacklist with sub-50https://bitcointalk.org/index.php?topic=312668.0, 2013.

[209] C. Natoli and V. Gramoli, "The balance attack or why forkable blockchains are ill-suited for consortium," in *Dependable Systems and Networks (DSN), 2017 47th Annual IEEE/IFIP International Conference on*. IEEE, 2017, pp. 579–590.

[210] C. Natoli and V. Gramoli, "The balance attack against proof-of-work blockchains: The r3 testbed as an example," *arXiv preprint arXiv:1612.09426*, 2016.

[211] C. Natoli and V. Gramoli, "The blockchain anomaly," in *Network Computing and Applications (NCA), 2016 IEEE 15th International Symposium on*. IEEE, 2016, pp. 310–317.

[212] I. Eyal, "The miner's dilemma," in *Security and Privacy (SP), 2015 IEEE Symposium on*. IEEE, 2015, pp. 89–103.

[213] A. Biryukov and D. Khovratovich, "Equihash: Asymmetric proof-of-work based on the generalized birthday problem," *Ledger*, vol. 2, pp. 1–30, 2017.

[214] M. Babaioff, S. Dobzinski, S. Oren, and A. Zohar, "On bitcoin and red balloons," in *Proceedings of the 13th ACM conference on electronic commerce*. ACM, 2012, pp. 56–73.

[215] J. A. Garay, A. Kiayias, N. Leonardos, and G. Panagiotakos, "Bootstrapping the blockchain-directly." *IACR Cryptology ePrint Archive*, vol. 2016, p. 991, 2016.

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

[217] A. Norta, "Creation of smart-contracting collaborations for decentralized autonomous organizations," in *International Conference on Business Informatics Research*. Springer, 2015, pp. 3–17.

[218] A. Yasin and L. Liu, "An online identity and smart contract management system," in *Computer Software and Applications Conference (COMPSAC), 2016 IEEE 40th Annual*, vol. 2. IEEE, 2016, pp. 192–198.

[219] M. Alharby and A. van Moorsel, "Blockchain-based smart contracts: A systematic mapping study," *arXiv preprint arXiv:1710.06372*, 2017.

[220] F. Idelberger, G. Governatori, R. Riveret, and G. Sartor, "Evaluation of logic-based smart contracts for blockchain systems," in *International Symposium on Rules and Rule Markup Languages for the Semantic Web*. Springer, 2016, pp. 167–183.

[221] S. Hu, C. Cai, Q. Wang, C. Wang, X. Luo, and K. Ren, "Searching an encrypted cloud meets blockchain: A decentralized, reliable and fair realization," in *2018 IEEE Conference on Computer Communications, INFOCOM 2018, Honolulu, HI, USA, April 16-19, 2018*, 2018, pp. 792–800.

[222] R. Cheng, F. Zhang, J. Kos, W. He, N. Hynes, N. Johnson, A. Juels, A. Miller, and D. Song, "Ekiden: A platform for confidentiality-preserving, trustworthy, and performant smart contract execution," *arXiv preprint arXiv:1804.05141*, 2018.

[223] K. Bhargavan, A. Delignat-Lavaud, C. Fournet, A. Gollamudi, G. Gonthier, N. Kobeissi, N. Kulatova, A. Rastogi, T. Sibut-Pinote, N. Swamy, and S. Zanella-Béguelin, "Formal verification of smart contracts: Short paper," in *Proceedings of the 2016 ACM Workshop on Programming Languages and Analysis for Security*. ACM, 2016, pp. 91–96.

[224] F. Zhang, E. Cecchetti, K. Croman, A. Juels, and E. Shi, "Town crier: An authenticated data feed for smart contracts," in *Proceedings of the 2016 aCM sIGSAC conference on computer and communications security*. ACM, 2016, pp. 270–282.

[225] A. Juels, A. Kosba, and E. Shi, "The ring of gyges: Investigating the future of criminal smart contracts," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 283–295.

[226] J. Krupp and C. Rossow, "teether: Gnawing at ethereum to automatically exploit smart contracts," in *27th USENIX Security Symposium, USENIX Security 2018, Baltimore, MD, USA, August 15-17, 2018.*, 2018, pp. 1317–1333.

[227] S. D. Lerner, "Drivechains, sidechains and hybrid 2-way peg designs," 2016.

[228] G. Maxwell, "Bringing new elements to bitcoin with sidechains," *SF Bitcoin Devs Meetup*, 2015.

[229] M. Pilkington, "11 blockchain technology: principles and applications," *Research handbook on digital transformations*, p. 225, 2016.

[230] O. Hueber, "The blockchain and the sidechain innovations for the electronic commerce beyond the bitcoin's framework," *International Journal of Transitions and Innovation Systems*, vol. 6, no. 1, pp. 88–102, 2018.

[231] P. Noizat, "Blockchain electronic vote," in *Handbook of digital currency*. Elsevier, 2015, pp. 453–461.

[232] J. Dilley, A. Poelstra, J. Wilkins, M. Piekarska, B. Gorlick, and M. Friedenbach, "Strong federations: An interoperable blockchain solution to centralized third party risks," *arXiv preprint arXiv:1612.05491*, 2016.

[233] S. Network, "Blockchain 4.0 is coming and seele is leading it," https://medium.com/@SummitNetwork/blockchain-4-0-is-coming-and-seele-is-leading-it-f88766fd9e32.

[234] K. Zhang and H. Jacobsen, "Towards dependable, scalable, and pervasive distributed ledgers with blockchains," in *38th IEEE International Conference on Distributed Computing Systems, ICDCS 2018, Vienna, Austria, July 2-6, 2018*, 2018, pp. 1337–1346.

[235] Dash, https://www.dash.org/.

[236] V. Buterin, "Ethereum: A next-generation smart contract and decentralized application platform, 2013," *URL {http://ethereum. org/ethereum. html}*, 2017.

[237] fabric, https://get.fabric.io/.

[238] Sawtooth, https://sawtooth.hyperledger.org/docs/core/releases/1.0/introduction.html.

[239] Iroha, https://cn.hyperledger.org/projects/iroha.

[240] J. L. Zhao, S. Fan, and J. Yan, "Overview of business innovations and research opportunities in blockchain and introduction to the special issue," 2016.

[241] J. Mendling, I. Weber, W. Van Der Aalst, J. v. Brocke, C. Cabanillas, F. Daniel, S. Debois, C. Di Ciccio, M. Dumas, S. Dustdar *et al.*, "Blockchains for business process management-challenges and opportunities," *arXiv preprint arXiv:1704.03610*, 2017.

[242] C. Prybila, S. Schulte, C. Hochreiner, and I. Weber, "Runtime verification for business processes utilizing the bitcoin blockchain," *Future Generation Computer Systems*, 2017.

[243] L. García-Bañuelos, A. Ponomarev, M. Dumas, and I. Weber, "Optimized execution of business processes on blockchain," in *International Conference on Business Process Management*. Springer, 2017, pp. 130–146.

[244] Y. Guo and C. Liang, "Blockchain application and outlook in the banking industry," *Financial Innovation*, vol. 2, no. 1, p. 24, 2016.

[245] L. Cocco, A. Pinna, and M. Marchesi, "Banking on blockchain: Costs savings thanks to the blockchain technology," *Future Internet*, vol. 9, no. 3, p. 25, 2017.

[246] D. Yermack, "Corporate governance and blockchains," *Review of Finance*, vol. 21, no. 1, pp. 7–31, 2017.

[247] N. Christin, "Traveling the silk road: A measurement analysis of a large anonymous online marketplace," in *Proceedings of the 22nd international conference on World Wide Web*. ACM, 2013, pp. 213–224.

[248] K. Soska and N. Christin, "Measuring the longitudinal evolution of the online anonymous marketplace ecosystem." in *USENIX Security Symposium*, 2015, pp. 33–48.

[249] R. Keyser, "Blockchain: A primer for governments," https://ujomusic.com/, 2017.

[250] S. Ølnes, "Beyond bitcoin enabling smart government using blockchain technology," in *International Conference on Electronic Government and the Information Systems Perspective*. Springer, 2016, pp. 253–264.

[251] M. Atzori, "Blockchain technology and decentralized governance: Is the state still necessary?" 2015.

[252] S. Ølnes, J. Ubacht, and M. Janssen, "Blockchain in government: Benefits and implications of distributed ledger technology for information sharing," 2017.

[253] K. Biswas and V. Muthukkumarasamy, "Securing smart cities using blockchain technology," in *High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016 IEEE 18th International Conference on*. IEEE, 2016, pp. 1392–1393.

[254] M.-L. Marsal-Llacuna, "Future living framework: Is blockchain the next enabling network?" *Technological Forecasting and Social Change*, 2017.

[255] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C. P. A. Ogah, and Z. Sun, "Blockchain-based dynamic key management for heterogeneous intelligent transportation systems," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1832–1843, 2017.

[256] C. Xu, K. Wang, and M. Guo, "Intelligent resource management in blockchain-based cloud datacenters," *IEEE Cloud Computing*, vol. 4, no. 6, pp. 50–59, 2017.

[257] T.-T. Kuo, H.-E. Kim, and L. Ohno-Machado, "Blockchain distributed ledger technologies for biomedical and health care applications," *Journal of the American Medical Informatics Association*, vol. 24, no. 6, pp. 1211–1220, 2017.

[258] M. Mettler, "Blockchain technology in healthcare: The revolution starts here," in *e-Health Networking, Applications and Services (Healthcom), 2016 IEEE 18th International Conference on*. IEEE, 2016, pp. 1–3.

[259] P. T. S. Liu, "Medical record system using blockchain, big data and tokenization," in *International Conference on Information and Communications Security*. Springer, 2016, pp. 254–261.

[260] V. Dhillon, D. Metcalf, and M. Hooper, "Blockchain in health care," in *Blockchain Enabled Applications*. Springer, 2017, pp. 125–138.

[261] X. Yue, H. Wang, D. Jin, M. Li, and W. Jiang, "Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control," *Journal of medical systems*, vol. 40, no. 10, p. 218, 2016.

[262] K. Peterson, R. Deeduvanu, P. Kanjamala, and K. Boles, "A blockchain-based approach to health information exchange networks," in *Proc. NIST Workshop Blockchain Healthcare*, vol. 1, 2016, pp. 1–10.

[263] Z. Shae and J. J. Tsai, "On the design of a blockchain platform for clinical trial and precision medicine," in *Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on*. IEEE, 2017, pp. 1972–1980.

[264] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the internet of things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.

[265] A. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, "On blockchain and its integration with iot. challenges and opportunities," *Future Generation Comp. Syst.*, vol. 88, pp. 173–190, 2018.

[266] S. Huh, S. Cho, and S. Kim, "Managing iot devices using blockchain platform," in *Advanced Communication Technology (ICACT), 2017 19th International Conference on*. IEEE, 2017, pp. 464–467.

[267] N. Kshetri, "Can blockchain strengthen the internet of things?" *IT Professional*, vol. 19, no. 4, pp. 68–72, 2017.

[268] A. Bahga and V. K. Madisetti, "Blockchain platform for industrial internet of things," *Journal of Software Engineering and Applications*, vol. 9, no. 10, p. 533, 2016.

[269] P. Ghuli, U. P. Kumar, and R. Shettar, "A review on blockchain application for decentralized decision of ownership of iot devices," *Adv. Comput. Sci. Technol*, vol. 10, pp. 2449–2456, 2017.

[270] M. Ruta, F. Scioscia, S. Ieva, G. Capurso, and E. Di Sciascio, "Semantic blockchain to improve scalability in the internet of things," *Open Journal of Internet Of Things (OJIOT)*, vol. 3, no. 1, pp. 46–61, 2017.

[271] A. Dorri, S. S. Kanhere, and R. Jurdak, "Blockchain in internet of things: challenges and solutions," *arXiv preprint arXiv:1608.05187*, 2016.

[272] A. Dorri, S. S.Kanhere, and R. Jurdak, "Towards an optimized blockchain for iot," in *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation, IoTDI 2017, Pittsburgh, PA, USA, April 18-21, 2017*, 2017, pp. 173–178.

[273] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in *Pervasive Computing and Communications Workshops (PerCom Workshops), 2017 IEEE International Conference on*. IEEE, 2017, pp. 618–623.

[274] J. H. Park and J. H. Park, "Blockchain security in cloud computing: Use cases, challenges, and solutions," *Symmetry*, vol. 9, no. 8, p. 164, 2017.

[275] K. Karlsson, W. Jiang, S. B. Wicker, D. Adams, E. Ma, R. van Renesse, and H. Weatherspoon, "Vegvisir: A partition-tolerant blockchain for the internet-of-things," in *38th IEEE International Conference on Distributed Computing Systems, ICDCS 2018, Vienna, Austria, July 2-6, 2018*, 2018, pp. 1150–1158.

[276] F. Montori, L. Bedogni, and L. Bononi, "A collaborative internet of things architecture for smart cities and environmental monitoring," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 592–605, 2018.

[277] P. Ta-Shma, A. Akbar, G. Gerson-Golan, G. Hadash, F. Carrez, and K. Moessner, "An ingestion and analytics architecture for iot applied to smart city use cases," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 765–774, 2018.

[278] M. Samaniego and R. Deters, "Hosting virtual iot resources on edge-hosts with blockchain," in *Computer and Information Technology (CIT), 2016 IEEE International Conference on*. IEEE, 2016, pp. 116–119.

[279] H. Shafagh, L. Burkhalter, A. Hithnawi, and S. Duquennoy, "Towards blockchain-based auditable storage and sharing of iot data," in *Proceedings of the 2017 on Cloud Computing Security Workshop*. ACM, 2017, pp. 45–50.

[280] Y. Zhang and J. Wen, "An iot electric business model based on the protocol of bitcoin," in *Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on*. IEEE, 2015, pp. 184–191.

[281] K. R. Ozyilmaz, M. Dogan, and A. Yurdakul, "Idmob: Iot data marketplace on blockchain," in *Crypto Valley Conference on Blockchain Technology, CVCBT 2018, Zug, Switzerland, June 20-22, 2018*, 2018, pp. 11–19.

[282] E. Fitzgerald, M. Pióro, and A. Tomaszewski, "Energy-optimal data aggregation and dissemination for the internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 955–969, 2018.

[283] J. Pan, J. Wang, A. Hester, I. AlQerm, Y. Liu, and Y. Zhao, "Edgechain: An edge-iot framework and prototype based on blockchain and smart contracts," *CoRR*, vol. abs/1806.06185, 2018.

[284] G. Pinto, J. P. Dias, and H. S. Ferreira, "Blockchain-based PKI for crowdsourced iot sensor information," *CoRR*, vol. abs/1807.03863, 2018.

[285] A. Manzoor, M. Liyanage, A. Braeken, S. S. Kanhere, and M. Ylianttila, "Blockchain based proxy re-encryption scheme for secure iot data sharing," *CoRR*, vol. abs/1811.02276, 2018.

[286] A. Boudguiga, N. Bouzerna, L. Granboulan, A. Olivereau, F. Quesnel, A. Roger, and R. Sirdey, "Towards better availability and accountability for iot updates by means of a blockchain," in *2017 IEEE European Symposium on Security and Privacy Workshops, EuroS&P Workshops 2017, Paris, France, April 26-28, 2017*, 2017, pp. 50–58.

[287] B. Liu, X. L. Yu, S. Chen, X. Xu, and L. Zhu, "Blockchain based data integrity service framework for iot data," in *2017 IEEE International Conference on Web Services, ICWS 2017, Honolulu, HI, USA, June 25-30, 2017*, 2017, pp. 468–475.

[288] C. Machado and A. A. M. Fröhlich, "Iot data integrity verification for cyber-physical systems using blockchain," in *21st IEEE International Symposium on Real-Time Distributed Computing, ISORC 2018, Singapore, Singapore, May 29-31, 2018*, 2018, pp. 83–90.

[289] O. J. A. Pinno, A. R. A. Grégio, and L. C. E. D. Bona, "Controlchain: Blockchain as a central enabler for access control authorizations in the iot," in *2017 IEEE Global Communications Conference, GLOBECOM 2017, Singapore, December 4-8, 2017*, 2017, pp. 1–6.

[290] O. Novo, "Blockchain meets iot: An architecture for scalable access management in iot," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1184–1195, 2018.

[291] S. Moon, H. Lee, and J. Lee, "SARA: sparse code multiple access-applied random access for iot devices," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 3160–3174, 2018.

[292] L. Zhou, L. Wang, Y. Sun, and P. Lv, "Beekeeper: A blockchain-based iot system with secure storage and homomorphic computation," *IEEE Access*, vol. 6, pp. 43 472–43 488, 2018.

[293] M. Wazid, A. K. Das, V. Odelu, N. Kumar, M. Conti, and M. Jo, "Design of secure user authenticated key management protocol for generic iot networks," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 269–282, 2018.

[294] R. Dennis and G. Owen, "Rep on the block: A next generation reputation system based on the blockchain," in *Internet Technology and Secured Transactions (ICITST), 2015 10th International Conference for*. IEEE, 2015, pp. 131–138.

[295] D. Carboni, "Feedback based reputation on top of the bitcoin blockchain," *arXiv preprint arXiv:1502.01504*, 2015.

[296] A. Lazarovich, "Invisible ink: blockchain for data privacy," Ph.D. dissertation, Massachusetts Institute of Technology, 2015.

[297] C. Jämthagen and M. Hell, "Blockchain-based publishing layer for the keyless signing infrastructure," in *Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), 2016 Intl IEEE Conferences*. IEEE, 2016, pp. 374–381.

[298] M. Baldi, F. Chiaraluce, E. Frontoni, G. Gottardi, D. Sciarroni, and L. Spalazzi, "Certificate validation through public ledgers and blockchains." in *ITASEC*, 2017, pp. 156–165.

[299] M. Ali, J. C. Nelson, R. Shea, and M. J. Freedman, "Blockstack: A global naming and storage system secured by blockchains." in *USENIX Annual Technical Conference*, 2016, pp. 181–194.

[300] D. Wilson and G. Ateniese, "From pretty good to great: Enhancing pgp using bitcoin and the blockchain," in *International Conference on Network and System Security*. Springer, 2015, pp. 368–375.

[301] S. Matsumoto and R. M. Reischuk, "Ikp: Turning a pki around with blockchains." *IACR Cryptology ePrint Archive*, vol. 2016, p. 1018, 2016.

[302] S. Muftic, "Bix certificates: Cryptographic tokens for anonymous transactions based on certificates public ledger," *Ledger*, vol. 1, pp. 19–37, 2016.

[303] R. Longo, F. Pintore, G. Rinaldo, and M. Sala, "On the security of the blockchain bix protocol and certificates," *arXiv preprint arXiv:1607.08401*, 2016.

[304] A. Savelyev, "Copyright in the blockchain era: Promises and challenges," *Computer Law & Security Review*, 2017.

[305] N. Fotiou and G. C. Polyzos, "Decentralized name-based security for content distribution using blockchains," in *Computer Communications Workshops (INFOCOM WKSHPS), 2016 IEEE Conference on*. IEEE, 2016, pp. 415–420.

[306] Y. Liu, K. Wang, Y. Lin, and W. Xu, "Lightchain: A lightweight blockchain system for industrial internet of things," *IEEE Transactions on Industrial Informatics*, 2019.

[307] A. B. Tran, X. Xu, I. Weber, M. Staples, and P. Rimba, "Regerator: a registry generator for blockchain," in *CAiSE Forum and Doctoral Consortium Papers*, 2017, pp. 81–88.

[308] X. Xu, C. Pautasso, L. Zhu, V. Gramoli, A. Ponomarev, A. B. Tran, and S. Chen, "The blockchain as a software connector," in *Software Architecture (WICSA), 2016 13th Working IEEE/IFIP Conference on*. IEEE, 2016, pp. 182–191.

[309] A. Pazaitis, P. De Filippi, and V. Kostakis, "Blockchain and value systems in the sharing economy: The illustrative case of backfeed," *Technological Forecasting and Social Change*, vol. 125, pp. 105–115, 2017.

[310] S. Huckle, R. Bhattacharya, M. White, and N. Beloff, "Internet of things, blockchain and shared economy applications," *Procedia Computer Science*, vol. 98, pp. 461–466, 2016.

[311] C. Dong, Y. Wang, A. Aldweesh, P. McCorry, and A. van Moorsel, "Betrayal, distrust, and rationality: Smart counter-collusion contracts for verifiable cloud computing," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 211–227.

[312] A. R. Choudhuri, M. Green, A. Jain, G. Kaptchuk, and I. Miers, "Fairness in an unfair world: Fair multiparty computation from public bulletin boards," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 719–728.

[313] S. Rowan, M. Clear, M. Gerla, M. Huggard, and C. M. Goldrick, "Securing vehicle to vehicle communications using blockchain through visible light and acoustic side-channels," *arXiv preprint arXiv:1704.02553*, 2017.

[314] B. Leiding, P. Memarmoshrefi, and D. Hogrefe, "Self-managed and blockchain-based vehicular ad-hoc networks," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. ACM, 2016, pp. 137–140.

[315] Q. Xing, B. Wang, and X. Wang, "Poster: Bgpcoin: A trustworthy blockchain-based resource management solution for bgp security," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 2591–2593.

[316] C. M. Matthews, "Silk road creator found guilty of cybercrimes," https://www.wsj.com/articles/silk-road-creator-found-guilty-of-cybercrimes-1423083107, 2015.

[317] Y. Cai and D. Zhu, "Fraud detections for online businesses: a perspective from blockchain technology," *Financial Innovation*, vol. 2, no. 1, p. 20, 2016.

[318] I. Weber, V. Gramoli, A. Ponomarev, M. Staples, R. Holz, A. B. Tran, and P. Rimba, "On availability for blockchain-based systems," in *Reliable Distributed Systems (SRDS), 2017 IEEE 36th Symposium on*. IEEE, 2017, pp. 64–73.

[319] B. M. in China, https://www.buybitcoinworldwide.com/mining/china/.

[320] C. Badertscher, J. Garay, U. Maurer, D. Tschudi, and V. Zikas, "But why does it work? a rational protocol design treatment of bitcoin," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2018, pp. 34–65.

[321] D. Aggarwal, G. K. Brennen, T. Lee, M. Santha, and M. Tomamichel, "Quantum attacks on bitcoin, and how to protect against them," *arXiv preprint arXiv:1710.10377*, 2017.

[322] M. T. Review, "Quantum computers pose imminent threat to bitcoin security," https://www.technologyreview.com/s/609408/quantum-computers-pose-imminent-threat-to-bitcoin-security/, 2017.

[323] L. Tessler and T. Byrnes, "Bitcoin and quantum computing," *arXiv preprint arXiv:1711.04235*, 2017.

[324] I. Stewart, D. Ilie, A. Zamyatin, S. Werner, M. F. Torshizi, and W. J. Knottenbelt, "Committing to quantum resistance: A slow defence for

bitcoin against a fast quantum computing attack," *IACR Cryptology ePrint Archive*, vol. 2018, p. 213, 2018.

[325] S. Popov, "The tangle," 2017.

[326] statista, "Internet of things (iot) connected devices installed base worldwide from 2015 to 2025 (in billions)," https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/.

[327] C. Dance, "Bitcoin nodes summary," https://coin.dance/nodes.

[328] I. Makhdoom, M. Abolhasan, H. Abbas, and W. Ni, "Blockchain's adoption in iot: The challenges, and a way forward," *Journal of Network and Computer Applications*, 2018.

[329] IOTA, "The iota vision," https://www.iota.org/the-foundation/our-vision.

[330] Wikipedia, "2016 dyn cyberattack," https://en.wikipedia.org/wiki/2016_Dyn_cyberattack.

[331] N. M. Kumar and P. K. Mallick, "Blockchain technology for security issues and challenges in iot," *Procedia Computer Science*, vol. 132, pp. 1815–1823, 2018.

[332] M. A. Khan and K. Salah, "Iot security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, vol. 82, pp. 395–411, 2018.

[333] M. T. Hammi, B. Hammi, P. Bellot, and A. Serhrouchni, "Bubbles of trust: A decentralized blockchain-based authentication system for iot," *Computers & Security*, vol. 78, pp. 126–142, 2018.

[334] M. Banerjee, J. Lee, and K.-K. R. Choo, "A blockchain future for internet of things security: A position paper," *Digital Communications and Networks*, vol. 4, no. 3, pp. 149–160, 2018.

[335] D. Rajan and M. Visser, "Quantum blockchain using entanglement in time," *arXiv preprint arXiv:1804.05979*, 2018.

[336] K. Ikeda, "qbitcoin: A peer-to-peer quantum cash system," *arXiv preprint arXiv:1708.04955*, 2017.

[337] A. Stanciu, "Blockchain based distributed control system for edge computing," in *Control Systems and Computer Science (CSCS), 2017 21st International Conference on*. IEEE, 2017, pp. 667–671.

[338] Y. Jiao, P. Wang, D. Niyato, and Z. Xiong, "Social welfare maximization auction in edge computing resource allocation for mobile blockchain," *arXiv preprint arXiv:1710.10595*, 2017.

[339] Z. Xiong, S. Feng, D. Niyato, P. Wang, and Z. Han, "Edge computing resource management and pricing for mobile blockchain," *arXiv preprint arXiv:1710.01567*, 2017.

[340] P. K. Sharma, M.-Y. Chen, and J. H. Park, "A software defined fog node based distributed blockchain cloud architecture for iot," *IEEE Access*, vol. 6, pp. 115–124, 2018.

[341] X. Sun and N. Ansari, "Edgeiot: Mobile edge computing for the internet of things," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 22–29, 2016.

[342] M. Swan, "Blockchain thinking: The brain as a dac (decentralized autonomous organization)," in *Texas Bitcoin Conference*, 2015, pp. 27–29.

[343] S. Omohundro, "Cryptocurrencies, smart contracts, and artificial intelligence," *AI matters*, vol. 1, no. 2, pp. 19–21, 2014.

[344] N. C. Luong, Z. Xiong, P. Wang, and D. Niyato, "Optimal auction for edge computing resource management in mobile blockchain networks: A deep learning approach," *arXiv preprint arXiv:1711.02844*, 2017.

[345] M. Laskowski, "A blockchain-enabled participatory decision support framework," in *Social, Cultural, and Behavioral Modeling - 10th International Conference, SBP-BRiMS 2017, Washington, DC, USA, July 5-8, 2017, Proceedings*, 2017, pp. 329–334.

[346] "Decentraland," https://decentraland.org/.

[347] I. 307, "Blockchain and distributed ledger technologies," https://www.iso.org/committee/6266604.html.

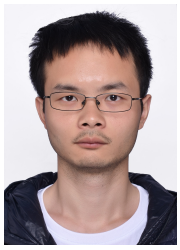
[348] "International harmonized stage codes," https://www.iso.org/stage-codes.html#20.00.

[349] "Blockchain reference architecture," http://www.cbdforum.cn/bcweb/index/article/bzwrr-1.html.

[350] "Blockchain data format specification," http://www.cbdforum.cn/bcweb/index/bz/1-0.html.

[351] "Standard for the framework of blockchain use in internet of things (iot)," http://standards.ieee.org/develop/project/2418.html.

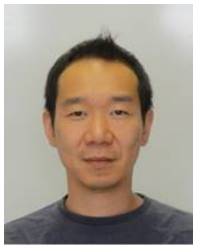
[352] M. Sporny and D. Longley, "The web ledger protocol 1.0," http://standards.ieee.org/develop/project/2418.html.



Mingli Wu received his master's degree from Shanghai Jiao Tong University, China, in 2018. He is currently a research assistant in the Hong Kong Polytechnic University. His research interests include blockchain, networking, and wireless network security.

Kun Wang (M'13-SM'17) received two Ph.D. degrees from Nanjing University of Posts and Telecommunications, China in 2009 and from the University of Aizu, Japan in 2018, respectively, both in Computer Science. He was a Postdoc Fellow in UCLA, USA from 2013 to 2015, and a Research Fellow in the Hong Kong Polytechnic University, Hong Kong, from 2017 to 2018. He is currently a Senior Research Professor in UCLA. His current research interests are mainly in the area of big data, wireless communications and networking, energy Internet, and information security technologies. He is the recipient of IEEE GLOBECOM 2016 Best Paper Award, IEEE TCGCC Best Magazine Paper Award 2018, IEEE TCBD Best Conference Paper Award 2019, and IEEE ISJ Best Paper Award 2019. He serves as Associate Editor of IEEE Access, Editor of Journal of Network and Computer Applications, and Guest Editors of IEEE Network, IEEE Access, Future Generation Computer Systems, Peer-to-Peer Networking and Applications, IEICE Transactions on Communications, Journal of Internet Technology, and Future Internet.

CAI Xiao Qing received the B.S. degree in computer science and technology from Northeast University, Shenyang, China, in 2017. She is currently pursuing the Ph.D. degree from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. Her interests include blockchain, cloud computing and resource management.



Song Guo (M'02-SM'11) received the PhD degree in computer science from the University of Ottawa and was a professor with the University of Aizu. He is a full professor with the Department of Computing, The Hong Kong Polytechnic University. His research interests include big data, cloud computing and networking, and distributed systems with more than 400 papers published in major conferences and journals. His work was recognized by the 2016 Annual Best of Computing: Notable Books and Articles in Computing in ACM Computing Reviews. He is the recipient of the 2017 IEEE Systems Journal Annual Best Paper Award and other five Best Paper Awards from IEEE/ACM conferences. He was an associate editor of the IEEE Transactions on Parallel and Distributed Systems and an IEEE ComSoc distinguished lecturer. He is now on the editorial board of the IEEE Transactions on Emerging Topics in Computing, the IEEE Transactions on Sustainable Computing, the IEEE Transactions on Green Communications and Networking, and the IEEE Communications. He also served as general, TPC and symposium chair for numerous IEEE conferences. He currently serves as an officer for several IEEE ComSoc Technical Committees and a director in the ComSoc Board of Governors. He is a senior member of the IEEE.

Minyi Guo (F'17) received the PhD degree in computer science from the University of Tsukuba, Tsukuba, Japan. He is currently a Zhiyuan chair professor with Shanghai Jiao Tong University, Shanghai, China. His research interests include pervasive computing, parallel and distributed processing, and parallelizing compilers. In 2007, he received the Recruitment Program of Global Experts and Distinguished Young Scholars Award from the National Natural Science Foundation of China. He is on the editorial board of the IEEE Transactions on Parallel and Distributed Systems and the IEEE Transactions on Computers. He is a fellow of the IEEE.

Chunming Rong is a professor and head of the Center for IP-based Service Innovation at University of Stavanger in Norway. His research interests include cloud computing, big data analysis, security and privacy. He is co-founder and chairman of the Cloud Computing Association (CloudCom.org) and its associated conference and workshop series. He is a member of the IEEE Cloud Computing Initiative, and co-Editor-in-Chief of the Springer Journal of Cloud Computing.