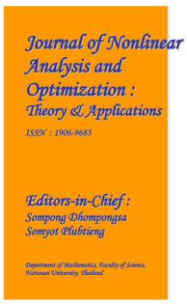
Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1, No.15 : 2024
ISSN : **1906-9685**



Paper ID: ICRTEM24_198 **ICRTEM-2024 Conference Paper**

TOWARDS DEPENDABLE, SCALABLE, AND PERVASIVE DISTRIBUTED LEDGERS WITH BLOCKCHAINS

[#1]CHITIKELA SRINIVAS, ***Assistant Professor,***
[#2]GOLI SUSHMA, ***Assistant Professor,***
Department of Computer Science & Engineering
GURUNANAK INSTITUTIONS TECHNICAL CAMPUS, IBRAHIMPATNAM

ABSTRACT: Distributed blockchain ledgers might disrupt numerous industries, including cryptocurrency. This disruption offers new public and commercial applications in supply chain management, financing, e-government, and e-health. Blockchain technology could enable decentralized data management. Blockchains offer auditability, security, trust, efficiency, and transparency because encrypted data is unchangeable. Distributed ledgers must overcome many obstacles before becoming widely utilized, stable, and scalable. Distribution ledger technology studies are examined in detail in this article. Blockchain generations, a rigorous blockchain application classification scheme, help achieve this goal. Comparisons between the CAP theorem and DCS (Decentralization, Consistency, and Scalability) are popular. Blockchain architecture has six layers: application, modeling, contract, system, data, and network. Finally, research viewpoints are categorized by DCS properties changed, applications examined, and hierarchical participation levels.

Keywords: *Blockchain, DCS, DLT, Cryptocurrency.*

1. INTRODUCTION

DLT, or blockchain, records and shares network-based transactions. Each user has access to the data, and cryptography links events, making modifications harder. Blockchain technology saves time, streamlines immutable document storage, verifies authenticity, and safeguards data. The previous solutions reduce duplicate data, fraud, abuse, and cybersecurity risks in systems and databases. Blockchain technology is famous for launching Bitcoin in 2008. Distributed Ledger Technology (DLT) will have a big influence. Modern systems like Hyperledger and Ethereum use enhanced computational capabilities to execute code, called Smart Contracts, to run several decentralized applications, called Apps. This technology could impact many industries, including the cryptocurrency industry. Government, healthcare, finance, and supply chain management employ IT. This enables app creation. Block chains are gaining popularity, but enterprise adoption is modest. Some individuals distrust this new technology and are unwilling to use it. Blockchain technology may be hard to integrate with enterprises' laboriously built regulatory processes and procedures. Without DLT platform certification or guarantees, applications may be subject to security and privacy concerns. The DAO Attack and Parity Multisig Bug add context. Given existing conditions, blockchain platforms cannot scale quickly enough to meet modern application needs.

These issues must be addressed by fundamental research in distributed systems, cryptography, and software engineering to maximize blockchain technology.

This document aims to systematically arrange and synthesize blockchain technology research. Because technology is continually changing, it's important to plan for future initiatives to improve DLT's adaptability. Developing a new framework can help blockchain professionals contextualize their work and assess its cross-domain applicability. Contrary to the proliferation of disorganized initiatives, blockchain technology needs comprehensive resources that explain its basics and operations. The first systematic academic blockchain research organization appears. Previously, research focused on contracts, security, and smart contracts. However, several studies have examined varied topics, including IoT growth and business process management. A comprehensive classification system for these investigations is still needed.

This paper adds these:

This portion explains how DCS—Decentralization, Consistency, and Scalability—affect the CAP theorem and clarifies crucial concepts.

Section 3 defines blockchain generations of blockchain applications.

Modern Distributed Ledger Technology (DLT) systems are six-layered blockchains, as described in Section.

Key research problems include targeted applications, linked layers, and affected DCS features (Section .

This article covers several crucial blockchain technology issues. This paper discusses blockchain's fundamental theories and components.

2. CORECONCEPTS

Financial transactions are detailed on distributed ledgers. New information is added to the ledger regularly. Many nodes store accurate log duplicates.

Key terms

Distributed ledgers consist of a blockchain data format, a peer-to-peer network of servers, and a consensus process that controls data addition. Based on its function, each component has its own possibilities. This diagram demonstrates a simple operational global ledger.

We can establish public and private distributed ledgers. Any peer-to-peer network participant can edit a public ledger. In contrast, a private ledger restricts access to a group of computers, usually corporations. Trust between nodes is the main difference between the ledgers. Nodes in a decentralized ledger system without peer-to-peer trust can operate independently. Thus, hostile actors seeking unauthorized system access need incentives or rewards. Private ledgers require peer confidence, sacrificing failure model resilience. Private ledgers provide more throughput and scalability than public ledgers, but they lose decentralization.

The following three sections outline distributed ledgers' three main components.

Data structure

Blockchain is the current distributed ledger data format. New recordings will be added to the book after a block is established. New blockchain blocks must be linked. A hash function verifies every previous block to protect current blocks from change. To replace an older block, all subsequent blocks must be modified, which is unlikely. Data in a block is valid based on its block age, or the number of subsequent blockchain blocks.

System component assembly varies widely. Merkle trees are hash-based. Merkle trees help lightweight clients find transactions quickly without a full log. Bitcoin uses Merkle trees for SPV. The figure above shows Bitcoin data format. Many people store their ledgers on blockchain due of its potential. Only blockchain-based ledgers will be examined.

P2Pnetwork

Peers share ledger data via networks. The network architecture of major blockchain systems is rarely disclosed. Every user in an unstructured overlay network has a unique neighbor group. Many peer-to-peer connections are needed to propagate new blocks and transactions.

Consensus

Blockchain enthusiasts and specialists are interested in the consensus algorithm. For the blockchain to work, peers use it to reach consensus on transactions for the next block. Executing smart contracts, choosing a leader, and defining mining incentives and transaction costs may require an agreement.

We use trustworthy professionals to generate consensus using facts.

Proof-based systems require block proposal and branch selection to attain consensus. The block proposition method lets any network peer offer a block. A peer can easily verify the block. Due to network latency, multiple blocks might be uploaded simultaneously, forming branches with different blockchain versions.

BlockchainPeer-to-Peer Network

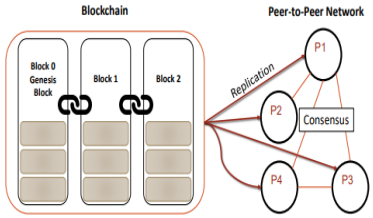


Figure. Core structure of distributed systems

Social group members choose a branch to unite diverse branches. This method ensures group understanding.

Satoshi Nakamoto invented Bitcoin's blockchain consensus technology. We employ Proof-of-Work block proposals to prevent data modification. The blockchain network must solve a computational problem to create a new block. Perfect peers are needed for Proof-of-Work and Nakamoto consensus procedures to continue developing the longest known branch. These two measures reduce blockchain data manipulation. An attacker would need more than 51% of the network's computing power to change current data and rewrite all following blocks to maintain consistency. Inactive client user transactions form blockchain blocks. The current blockchain, ending at block N, can generate several proof-containing blocks. This is due to slow networks and many processes. Finally, the longest chain method will require peers to choose one block and reject the others and agree on blockchain status.

Due to the high computing costs of the Proof-of-Work consensus mechanism, which hurt the economy and environment, many alternatives have been proposed. Proof-of-Stake powers Peer Coin. Contribute bitcoins to block creation. This simplifies computer security.

Running public ledgers requires people to join and add transactions to the blockchain. Bitcoin miners get paid for network transactions. Miner may receive the entire block reward in the current block and the next block. Bitcoin-NG uses proof- and leader-based consensus. The following supervisor recommends adding blocks, citing the Proof of Work. Because participants trust each other, private ledgers often use leader-based agreements. An ordering service sequences Hyperledger events. This procurement service is decentralized (alternating leaders) or centralized. The ordering service controls block proposal, therefore branch selection is unneeded. These committing peers must use PBFT to agree on outcomes as the ordering service does not execute transactions. Next are order or blocks.

Smart contracts

Smart contracts for non-cryptocurrencies are new. Certain smart contracts are executed automatically by blockchain miners. Smart contracts are transparent because they are reviewed and approved by the public before deployment to the blockchain. Because blockchains are unchangeable, smart contracts must be precise. Hyperledger and Ethereum are popular smart contract technologies.

Solidity-based Ethereum Hello World:

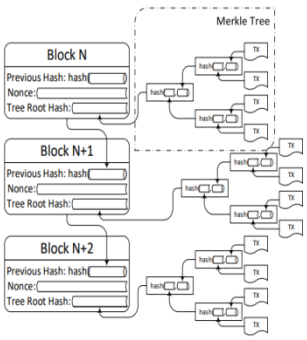


Figure. Block Layout

```
pragma solidity \begin{lstlisting}^0.4.4;
contract HelloWorld {
string public greeting;
function HelloWorld(string _greeting) {greeting=_greeting;
}
function setGreeting(string _greeting) {greeting=_greeting;
}
```

uses cryptography for security and privacy.

```
function say() constant returns (string) { in applications running in environments
where the network
return greeting;
}
}
```

setGreeting and Hello World produce new messages and objects, respectively. Gas is given to miners who add transactions to these blocks. As a constant function that reads previously read data, Say() requires no fuel. This allows it to function without a transaction.

Distributed databases

Subclasses of DDBMSs include blockchain systems. Many large web apps use DDBMS. Hase, Cassandra, Mongo DB, Oracle, IBM, and many others offer no SQL key-value stores, sharded relational databases, and novel SQL data stores. Blockchain administration is decentralized, therefore users may be competitors or strangers. They prefer their own data integrity proofs over external ones. Block chains can lower the cost and complexity of building a shared DDBMS by simplifying crucial document interchange. Blockchain systems are slower than DDBMSs because privacy and security need more processing power. More research is needed to compare blockchain systems to other data systems without compromising security or privacy. Many share files via BitTorrent and Freenet. These censorship-resistant, decentralized networks lack content upload consensus, unlike Bitcoin.

DCS properties

Blockchain systems aim to retain these three properties:

Decentralization: Blockchains may be relied on without a third party due to their decentralization. They forbid data and app restriction. When confidence is lacking, blockchain data must remain secret and anonymous.

Consistency: Data on a blockchain should be uniform. We can't undo past mistakes. The blockchain's history should verify its health.

A single company stores data in multiple data centers using DDBMS. Blockchains store data centrally.

Blockchain searches should yield peer-independent results.

Scalability: As users and processing capacity increase, system availability and performance (throughput and latency) should improve. It should handle more users, smart contracts, and inquiries.

The CAP theorem states that a blockchain system can only provide two of the three properties simultaneously. We demonstrate this with three cases.

To maintain the blockchain, Bitcoin is a decentralized system (DC). The incentive-driven Proof-of-Work network makes Bitcoin unscalable. To mine more blocks, miners gradually increase hash power. The ten-minute Bitcoin mining process produces seven transactions per second despite being more powerful.

Its stable public ledger makes Ethereum a distributed ledger. DCS characteristics are a spectrum, like the first CAP theorem. Ethereum's block time is faster than Bitcoin at 10-40 seconds, down from 10 minutes. Thus, branch frequency increases, threatening regularity. Ethereum GHOST branch selection mitigates this problem.

Because permissioned networks are better than anarchy, Hyperledger is CS. Proof-of-Work is replaced by a fast ordering service that can process over 10,000 events per second in Hyperledger.

Changes to the blockchain generally increase some functions and decrease others. Blockchain applications are one size does not fit all. DCS features can be added to blockchain systems for several use applications.

3. BLOCKCHAINAPPLICATIONG ENERATIONS

Distributed Consensus Systems (DCS) enable all blockchain use cases and applications. Sorting

anticipated use cases into discrete groups simplifies choosing research projects to implement the desired applications.

According to this report, Blockchain has three stages: 1.0, 2.0, and 3.0. Time establishes generational succession, but new generations do not destroy their predecessors. Actually, all three program versions are being modified simultaneously. It is believed that each generation introduces new issues and exciting research ideas.

Blockchain1.0:Cryptocurrency

Blockchain 1.0 uses crypto. The distributed system tracks wallet-to-wallet digital asset transfers. System transfer validation is hardcoded.

The industry supports Blockchain 1.0, the initial blockchain technology. Around 600 cryptocurrencies are publicly available, including Bitcoin.

Bitcoin is supposed to replace fiat cash, even though 1.0 apps are used for investments. To do this, blockchain systems must be more scalable, decentralized, and reliable. To maintain legacy systems like Bitcoin, modifications and additions must be made gradually. Because new blockchain code is incompatible with old code, hard forks are worrying. When users resist code changes, a hard branch may split the group.

Currently, 1.0 applications are utilized for investments, but cryptocurrency will eventually replace fiat cash. To do this, 1.0 blockchain systems must be more dependable, scalable, and decentralized. Changes must be gradual to support older systems like Bitcoin. Problematic are hard forks, which occur when new blockchain code doesn't operate with old. When users resist code updates, a hard fork splits them.

Blockchain2.0:ĐApps

Smart contracts help decentralized applications run on Blockchain 2.0. Blockchain 2.0 application platforms have many smart contract languages and a public cloud for code storage. Ethereum developers can store apps in contract accounts, but only after paying a one-time cost for contract code storage. After triggering smart contracts, Ethereum users can access any application. The miner receives gas for the transaction fee when the code is executed.

Applications 2.0 set their own digital asset exchange rules, unlike cryptocurrencies. Crowdsourcing, DAOs, prediction, and charity are examples.

Research should prioritize smart contract security study for version 2.0. These contracts must be rigorously verified before being permanently added to the blockchain. Version 2.0 has revolutionary scaling methods.

A score of 1.0 is reasonable given smart contracts and operating costs.

Blockchain3.0: Pervasive apps

Governments and corporations use Blockchain 3.0. Applications over 3.0 require private infrastructure because they are too large for public clouds. Internet of Things-enabled 3.0 apps are expected to interact with the physical environment. Land registries, supply chain management, and eHealth are examples.

Built-from-the-ground-up blockchain solutions may be more scalable than 3.0 applications because maintenance takes time. Application use cases limited to a few enterprises may hinder decentralization. This enhances consistency and scalability. Hyperledger and Corda are third-generation blockchains.

4. BLOCKCHAINLAYERS

The diagram shows the published peer-to-peer blockchain networking and application reference architecture. Layer difficulties are given in ascending order.

Application layer

Develops blockchain solutions for specific industries and applications. Creating exact blockchain technology specifications from stakeholder application requirements is an intriguing research field. Studying successful and unsuccessful blockchain technology stories is essential to understanding blockchain researchers and developers' challenges. Feasibility studies that assess blockchain technology and end-user needs are needed to choose the optimum blockchain system design. The above goals motivate blockchain technology's improvement, expansion, and possibly redefining. Showing blockchain

technology's benefits and applicability with application prototypes. Research is needed to determine the application parameters that will affect blockchain deployment.

Modeling layer

Work flow models are needed to build intelligent contracts that enforce application semantics. Understanding the technology layers of sophisticated blockchain modeling languages like BPMN requires a thorough investigation. These blockchain integration concepts can characterize new and existing applications. Additionally, blockchain deployment methods can be improved. Security and privacy conventions must be implemented to guarantee lower levels comply. Smart contract execution modeling software may now be standardized.

Contract layer

The development of smart contracts is early. These technologies must prove verifiability, dependability, and security before widespread usage. Poor smart contracts can cause financial losses, thus they must be tested and confirmed before being placed on a live blockchain. Programming languages must have strong security features to reduce security risks. Smart contracts express and combine reusable services and middleware.

System layer

System layer includes blockchain consensus mechanism and subsystems. Research struggles with leadership and proof algorithms' consistency and scalability. The system must maintain secrecy to facilitate sensitive data applications.

Data layer

Data layers store block-level on-chain and off-chain data in databases. Ethereum and Hyperledger provide off-chain data storage to reduce blockchain data and peer strain from preserving complete blockchain copies. This reduces the quantity of data on the blockchain and relieves peers, who must keep entire copies. Off-chain data stability may be unreliable. Data storage issues include security, privacy, and performance, whether on-chain or off-chain. The hardest part is deciding whether data goes on-chain or off-chain.

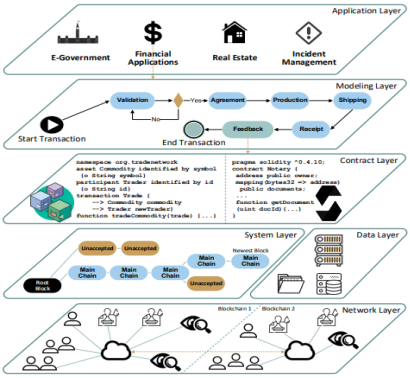


Figure3. Blockchain Stack refers to many blockchain technology and components.

Network layer

Blockchains require network access to share status updates. Network conditions affect blockchain efficiency, security, and privacy, hence they must be examined. Existing methods often miss and underspecify blockchains' aspect. More research is needed to find the best network cross-blockchain communication technique to increase block chain interoperability and cross-application interactions.

Main Research Challenges

In this context, blockchain research is grouped by impacted DCS elements, planned applications, and layers.

Applicability of blockchains

Case studies in impoverished countries could help popularize blockchain technology. Developing a use case research methodology. This is the preferred use case description format:

The user wants academic rewriting. The user, however,

How was it fixed?

These people use actors:

Who transmits transactions?

Who creates intelligent contracts?

The industry's most famous and respected actors?

Certain entities or individuals maintain and validate blockchains and transactions.

Who searches data without customers submitting transactions?

Examples of knowledge

Blockchains store varied data.

What are blockchain smart contracts?

What investigative powers can blockchain offer?

Academics study object-actor relationships.

Who creates, applies, and discloses an object?
To originate, access, and execute transactions, actors need what authorizations?
Who assigns item responsibility?
Detail performance specifications:
Looking for what kind of actor?
What is estimated transaction volume?
What delays can be expected?
What kind of system expansion is expected?
This study effort seeks the best blockchain platforms and apps.
This will not immediately affect DCS assets.
It focuses on the least-known 3.0 applications. Apps 2.0 with smart contracts have further benefits. Version 1.0 programs are well-tested, well-known, and limited.
Using modeling and application layers, this study examines numerous utilization situations. To fully comprehend blockchain systems and whether they are suitable for specific use cases, one must grasp their underlying layers.

Blockchain middleware

Nearly identical domains and apps. Reusable blockchain middleware boosts blockchain application efficiency. Microsoft Blockchain on Azure (based on Ethereum), Deloitte Rubix (based on a customized blockchain architecture), and IBM BlueMix are the top blockchain middleware solutions. These projects provide consumers with a managed blockchain infrastructure to enable Blockchain-as-a-Service (BaaS) by minimizing deployment resource requirements. These attempts may abstract our perspective's lower and system/data levels. We offer reusable components, modeling, and contract layer services to accelerate development and improve system reliability. At present, only one incidence is recorded.
Blockchain middleware should support messaging, event notification, identity management, data integration (especially sensor-based data integration), and analytics. The Application Blockchain Interface (ABCI) lets apps use the blockchain technology to reduce errors by duplicating state across devices. This will not affect DCS assets immediately. Middleware compatibility with Blockchain 2.0 and 3.0 is growing, which is crucial. 3.0 applications that interact with the real environment need data integration services that consider physical restrictions. Blockchain data must be protected from tampering or incorrect data transfer from real-world sensors. Off-chain payment networks and cross-platform cryptocurrency exchanges may be needed for 1.0 applications.
Intelligent contracts will dominate contract-layer blockchain middleware. Some middleware can be multiple-level pluggable subsystems.

Securityandprivacy

Security and privacy concerns are slowing blockchain adoption. Bitcoin transactions are encrypted, ensuring anonymity. It's vital to note that the blockchain records every transaction, allowing behavior-based tracking. Bitcoin's fungibility is threatened by fraudulent token addresses. Clean coins with a short transaction history command a small premium over their elder counterparts. Modern systems employ mixer networks to hide transaction records for privacy.
Since contract code is difficult to maintain private while validating transactions on a public network, smart contract systems present privacy issues. Cryptography like Zero Knowledge Proofs (zk-SNARK) can achieve this. Certain industrial applications may not meet privacy standards with data encryption. Data must be assured to conform with specific boundaries to meet legal requirements. In some cases, the blockchain platform must balance consistency and privacy. A multi-channel strategy is another option.
Finally, smart contract security must be upgraded to expand use. Smart contract validation tools are needed to find and fix errors. New languages may increase smart contract code readability and legality.

Scalable system innovations

Decentralization and consistency are needed to secure blockchain systems from breaches. As mixer networks demonstrate, scalability may be affected by enhancements that demand a lot of processing power or latency.
This topic affects generations differently. Version 2.0 emphasizes smart contracts' reliability and credibility, while Version 3.0 protects data

privacy. However, Version 1.0 prioritizes transaction privacy and traceability.

The contract, system, and data levels must be examined for this issue. The network layer is needed to integrate onion networks.

Major system development advances.

The adoption of blockchain technology depends on its scalability. The consensus method and its replacement, the scalable and eco-friendly Proof-of-Stake mechanism, are under study.

Blockchains that develop linearly and peers process transactions sequentially limit network capacity. A full system replication method is needed for blockchain. Parallelism like side chains and sharding improves system performance.

Lightning network transactions demonstrate another option: outsourcing transactions outside of blockchain technology.

Each peer's main memory blockchain state develops with users and smart contracts. To speed query response and transaction validation, new data structures and management methodologies must be tried. Merkle Patricia trees and IAVL+ trees are popular in computer science. As the blockchain ecosystem grows, a better protocol that lets new miners start without buying the blockchain is needed.

We must conclude that new technologies improve system performance. Intel Software Guard Extensions provides Hyperledger Sawtooth's Proof-of-Elapsed-Time consensus method.

Scalability improvements usually affect consistency or decentralization. Unlike permission-free systems, permissioned systems cannot decentralize completely. Because permissioned systems require branch merging or splitting. Data accessibility may be compromised by smart contracts or blocks without peer storage.

Solution viability is low, but scalability helps all three generations. The protocol re-parametrizations of Segwit2x for Bitcoin suggest small enhancements. 2.0 systems must be decentralized for public cloud networks. A minimum rating of 3.0 allows systems to operate without preconditions and use the best designs efficiently.

The consensus procedure is key to this topic, hence the system layer is crucial. Scalability affects the data layer.

5. CONCLUSIONS

Distributed ledger technology can alter several industries globally. To employ scalable, reliable, and widespread systems widely, basic research in many fields is needed.

This study employs a comprehensive blockchain research method. First, we discussed DCS (Decentralization, Consistency, and Scalability) and how blockchain systems must manage their connection. Blockchain applications have three generations: 1.0 for cryptocurrencies, 2.0 for other applications, and 3.0 for widespread ones. The layered graphic shows a blockchain architecture's Framework, Application, Modeling, Contract, System, Data, and Network levels. We proposed a blockchain technology study plan concentrating on middleware, security and privacy, scalable system developments, and applications.

To conclude, different application generations require different solutions.

Scholars can study the blockchain stack's layers in distributed systems, software engineering, networking, data administration, and cryptography.

Distributed consensus systems (DCS) and the CAP theorem are characterized by trade-offs. Thus, use cases are crucial for shaping blockchain system architecture to achieve the greatest combination of features for the application's aims. Blockchain technology does not have a universal solution.

REFERENCES

1. S.Nakamoto,Bitcoin:Apeer-to-peerelectroniccashsystem,2008,
2. URL:http://www.bitcoin.org/bitcoin.pdf,2012.
3. HyperledgerWhitepaper,http://www.the-blockchain.com/docs/Hyperledger%20Whitepaper.pdf,may2017.
4. G. Wood, Ethereum: A secure decentralisedgeneralised transactionledger,2017.Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, Hawk:The blockchain model of cryptography and

privacy-preserving smartcontracts, in Security and Privacy (SP), 2016 IEEE Symposium on.IEEE,2016,pp.839–858.

5. E.Hildenbrandt,M.Saxena,X.Zhu,N.Rodrigues ,P.Daian,D.Guth,and G. Rosu, Kevm: A complete semantics of the ethereum virtualmachine,Tech.Rep.,2017.
6. N.Atzei,M.Bartoletti,andT.Cimoli,Asurveyofa ttacksonethereumsmartcontracts(sok),inInterna tionalConferenceonPrinciplesofSecurityandTr ust.Springer,2017,pp.164–186.
7. L. S. Sankar, M. Sindhu, and M. Sethumadhavan, Survey of con- sensusprotocolsonblockchainapplications,inAd vancedComputingand Communication Systems (ICACCS), 2017 4th International Con-ferenceon. IEEE,2017,pp.1–5.
8. I.-C. Lin and T.-C. Liao, A survey of blockchain security issues andchallenges,IJNetworkSecurity,vol.19,no.5, pp.653–659,2017.Dorri, S. S. Kanhere, and R. Jurdak, Blockchain in internet ofthings: challenges and solutions, arXiv preprint arXiv:1608.05187,2016.
9. K.Croman,C.Decker,I.Eyal,A.E.Gencer,A.Juel s,A.Kosba,
10. Miller, P.Saxena, E.Shi,E. G.Sireretal.,On scalingde-centralized blockchains, in International Conference on FinancialCryptographyandDataSecurity.Sprin ger,2016,pp.106–125.