Copyright © 2018 Tech Science Press techscience, vol.1, no.1, pp.1-16, 2018

GaiaWorld: A Novel Blockchain System Based on Competitive PoS Consensus Mechanism

Ziming Liu[1], Min Tang[1], Rui Song[2] and Yubo Song[2]

Abstract: The birth of blockchain has promoted the development of electronic currencies such as Bitcoin and Ethereum. Blockchain builds a financial system based on cryptology instead of credit, which allows parties to complete the transaction on their own without the need for credible third-party intermediaries. So far, the application scenario of blockchain is mainly confined to the peer-to-peer electronic financial system, which obviously does not full utilize the potential of blockchain.

In this paper, we introduce GaiaWorld, a new system for decentralized application. To solve the problem of resource waste and mismatch between nodes and computing power in traditional PoW mechanism, GaiaWorld introduces a new consensus mechanism called CPoS, which can improve productivity and liquidity of blockchain system. GaiaWorld constructs a new architecture based on forging committee and forging group systems, which can establish a decentralized, free and stable internet trust system, and can be utilized in multiple application scenarios and construct efficient and reliable content delivery systems.

Keywords: Blockchain, Cryptocurrency, Consensus mechanism.

1 Introduction

Blockchain is originally defined as a mechanism for distributed accounting and data consistency[Nakamoto (2008)]. Bitcoin takes advantage of a peer-to-peer network to utilize a decentralized ledger system, where the account books record the verified transactions and encrypted them into blockchains. Under this decentralized system, trusted third-party entities can be abandoned during currency transactions[Ron and Shamir (2013)]. By introducing cryptography technologies, blockchain systems can achieve high stability and robustness in untrusted network.

After achieving big success in electronic currency fields, researches tend to apply blockchain technology to other fields. Ethereum initiated by Vitalik Buterin in 2014 is the most successful attempt[Buterin (2013)]. The biggest innovation of Ethereum is creating a block with perfect function and built-in Turing complete programming language

[1] Gaia.World Foundation, Chengdu, Sichuan, P. R. China

[2] School of Cybersecurity, Southeast University, Key Laboratory of Computer Network Technology of Jiangsu Province, Nanjing, Jiangsu, P. R. China

techscience. doi:10.32604/techscience.2018.??? www.techscience.com/techscience

which can be deeply and free regulated[Wood (2014)]. It allows customer to compile any complicated contract, autonomous agent and relation which are totally existed in blockchain and delivered by it. Users of Ethereum can achieve any type of transaction by the built-in script in the contract, including customized currency, financial derivatives, identity system or even more complicated decentralized application.

The core feature of Ethereum is the so-called 'smart contract', which is an encrypted box which includes value and can only be opened after meeting some specific conditions[Buterin (2014)]. Since the block is embedded with Turing complete programming language and has features like value-awareness, blockchain-awareness and multi-states, it is easy for Ethereum to take advantage of smart contract to achieve applications unimaginable in the past.

In decentralized systems, multiple hosts form network clusters through asynchronous communication. In such a distributed computing architecture, state consensus should be reached between hosts. Given the complexity of network and the existence of evilness nodes, fault tolerant protocol and consensus mechanism should be defined to realize credible data transmission and transaction negotiation systems[Vukolić (2015)].

Both Bitcoin and Ethereum use PoW (Proof of Work) to solve the consensus problem in distributed system. PoW is a consensus mechanism based on computing power pricing[Nakamoto (2008)]. The main feature of PoW systems is the asymmetry of calculation, that is, the calculation of the work itself is far more difficult than that of the verifier to verity that the work is completed correctly. Each node compete with each other to obtain the right of bookkeeping, and the winner receives a certain amount of reward. In Bitcoin system, a round of computing power competition begins every 10 minutes. The winner can build block and broadcast the block to the entire network.

However, the PoW mechanism has obvious disadvantages. On the one hand, PoW mechanism provide consistency and correctness on the premise that the computing nodes and their computational capabilities roughly match. But with the help of FPGA and ASIC, miners can achieve higher computing power which makes the number of nodes and the computing power lose the match. On the other hand, the tasks used to achieve consensus are extremely complex cryptography problems with no practical or scientific value, which is obviously a huge waste of computing and natural resources[Gervais, Karame, Wüst et al. (2016)].

In view of this, researchers have proposed some alternatives of PoW. PoS (Proof of Stake) is one of the methods which has been widely used. PoS is a new consensus mechanism based on on-chain currency pricing[Bentov, Lee, Mizrahi et al. (2014)]. PoS replaces the competition of computing power with cash holding. Nodes can get the right to bid for block generation by consuming certain interests. The **Coin Days** defines the possibility of winning, which is the product of amount of currency hold by the node and the holding time. The difficulty of mining will decrease with the increase of the Coin Days. PoS address the problem of excessive resources consumption of PoW mechanism. However, the difficulty of mining is related to the Coin Days of miners, which increases the possibility of fork.

PoS encourages users to hoard Coin Days, which reduces the liquidity of currency[King and Nadal (2012)].

In this paper, we introduce GaiaWorld, a new blockchain system for decentralized applications which puts forward a new PoS scheme: PoS based on competition (CPoS). Based on forging committee mechanism, CPoS can solve the problem of rich people getting richer easily in traditional PoS mechanism and improve the productivity and liquidity of the system. Based on CPoS, GaiaWorld can reconstruct most of traditional applications in decentralized manner and improve the performance of these applications. In addition, the super sidechain introduced by GaiaWorld provides developers with a secure and isolated developing environment and help developers build complex blockchain applications.

The rest of the paper is organized as follows. Section 2 presents the issues in traditional blockchain systems. Section 3 introduces the CPoS consensus mechanism and core technology of GaiaWorld. Section 4 performs the evaluation towards GaiaWorld which tests the performance of CPoS and forging committee in multiple scenarios. Section 5 discusses the prospect of on-chain applications and provide several possible application scenarios. Section 6 concludes the whole paper finally.

2 Technical Issues Faced by Blockchain

2.1 Performance Issue and Resources Waste

Though Ethereum greatly expands the application of blockchain by smart contract, the application scope is still restricted by several issues. The first one is performance issue caused by Proof of Work (PoW) consensus mechanism. Blockchain uses hash function to validate data[Biryukov and Pustogarov (2015)]. Specifically, hosts around the world compete to find out the nonce which can match the originally packaged data, and the winner gets the right to pack the block, which is the right of account charging. Given the performance growth proven by Moore's Law, it's usually set up to increase the difficulty of nonce in competition to maintain its reasonable operating speed[Shi (2016); Lin (2018)].

However, since the probability to get reward is proportional to computing power, all nodes will inevitably keep enhancing their computing power to win the competition. Thus most computing power is wasted in meaningless Hash computation, which leads to great waste of computing and energy resources. Statistics indicates that the electric energy cost in mining has exceeded the total amount of that in a small country currently[Omohundro (2014)].

2.2 High Transaction Cost

The transaction fee involved in Ethereum blockchain will be finally calculated with Ether. There shall be a certain amount of 'Gas' for each transaction and fees necessary to pay for each unit of Gas[Chen, Li, Wang et al. (2017)]. All operations during transaction execution, including database reading and writing, message sending, and calculation will consume a certain amount of Gas. Though in the design of Ethereum, the amount of Gas needed to pay for each transaction is fixed, the fees of each unit of Gas is still be designed dynamically by users[Kim (2017)]. Meanwhile, the higher transaction fee to be paid, the more active for nodes to pack and handle this transaction. Currently, to conduct the simplest Ether transfer transaction, 0.1 Ether should be paid as the transaction fee, which makes small volume of Ether transaction impossible[Sergey and Hobor (2017)].

2.3 Sidechain Interaction and Isolation

Another urgent issue that needs to be addressed is the sidechain. Sidechain is a fork generated based on the main blockchain to realize some specific purposes or functions. Before the birth of Ethereum, forks was usually performed by means of hard fork[Pilkington (2016); Christidis and Devetsikiotis (2016)]. The hard fork is essentially a set of sidechain derived by upgrading or adjusting codes based on a time node of original blockchain network. Though the new blockchain is homologous as the original blockchain, it's totally independent from the mainchain and cannot communicate with the mainchain.

Ethereum makes use of the feature of smart contract to design ERC20 token specification. Tokens based on Ethereum are essentially smart contracts running in main blockchai in a virtual state, which is different from the hard fork of BTC. As to simple token issuing, smart contract on Ethereum is a simple and efficient idea. But for some more complicated applications, running in smart contract in main blockchain will lead to a serious safety loophole. Since the sidechain in the form of smart contract and main blockchain are not isolated compulsorily as BTC sidechain, the defect in sidechain design may directly affect the mainchain[Xu, Pautasso, Zhu et al. (2016); Swan (2015); Norta (2015)].

Besides safety factor, the derived sidechain with smart contract is essentially a program running in Ethereum, which will inevitably further increase the volume and complexity of blockchain. It makes the situation worse for the whole Ethereum node network[Jentzsch (2016); Aitzhan and Svetinovic (2016)].

2.4 Difficulty to Interact with Applications

In Ethereum, there are two entities to initiate and receive transactions: users and smart contract. Smart contract can be regarded as an automatic agent in Ethereum network. It has Ethereum address and account sum, and can send and accept transactions. When there is someone sending transactions to the contract, it will be activated and starts to run its own programs, such as to change its internal status or send some transactions[Hirai (2017); Saito and Yamada (2016)].

The biggest problem of users in smart contract designing is that codes running in virtual machine cannot visit and call data outside of the blockchain network[Huh, Cho and Kim (2017)]. In actual process, the main challenge is that most derivative contracts need to be combined with a contract specially used for data release. But this needs to rely on some special agencies to regularly maintain and update data, and provide an interface to allow other contracts to send query message to gain key data.

3 Core Technology of GaiaWorld

3.1 Consensus Mechanism

Proof of work and Proof of stake are two most important concepts in blockchain technology. Blockchain is essentially a distributed ledger, so it inevitably faces two issues below:

1. How to build the concept of time sequence in decentralized network?
2. Whose records should be adopted when multiple nodes have completed record transactions?

Most of the blockchain applications use PoW (Proof of Work) to solve the consensus problem in distributed system[Zhu, Guo, Gan et al. (2016); Zhang, Cecchetti, Croman et al. (2016)]. To avoid the problems we mentioned in section 2, we introduce the PoS (Proof of Stake) mechanism, which is a new consensus mechanism based on on-chain currency pricing[Zheng, Xie, Dai et al. (2017)]. PoS replaces computing power with cash holding[Zikratov, Kuzmin, Akimenko et al. (2017); Xu, Xiang and Sachnev (2018)]. It enables cash-holders to participate more in the mining process and do not need to calculate complicated math problem, thus to avoid resource and energy waste. There are mainly 4 existing PoS solutions: PoS based on Byzantine fault tolerance, PoS based on chain, PoW/PoS mixture and Delegated Proof of Stake (DPoS)[Chen and Zhu (2017); Watanabe, Fujimura, Nakadaira et al. (2016, 2015)]. After studying existing PoW and PoS mechanism, GaiaWorld puts forward a new PoS scheme: PoS based on competition (CPoS).

3.1.1 CPoS Forging Committee

GaiaWorld generates and allocates 2.1 billion FBC in Genesis Block. Block creation is then completed by Forging Committee. To solve the common problem of rich people getting richer easily in PoS mechanism, except for genesis block, creation of other blocks will not generate new coins. All revenue come from transaction fees. Forging committee is a smart contract, including several committee member nodes which have forging right. To encourage forging, the member who successfully forges a block will gain all transaction fees in the block.

Every node can apply to join the forging committee with at least 1 FBC as a deposit. The responsibility of the committee members is to create new blocks. If a member node fails to perform forging obligation for continuously 3 times, it will be forced out of the committee, and the deposit will be withheld for a certain time.

3.1.2 Forging Group

The voting right of forging committee member is related to the sum of deposit. A newly joined committee member will not get the voting right immediately. It needs to wait for

100,000 new blocks before gaining the voting right. With the increase of block height, the voting right will be continuously accumulated. If a member successfully adds one block to blockchain, its voting right will be reset to 0. The members are divided into different forging groups according to the last two digits of its address. The member who gets the highest voting right in one group will be elected to be the main forging committee member. The successive forging groups tend to validate and recognize the block forged by main members.

The reward of forging committee members comes from all transaction fee in the block he newly creates and all deposit of the evil node reported by him. Due to little resources are consumed during block creating in CPoS system, members can get considerable profit even though there is just transaction fee as reward. Thus, negative side effects of the rich getting richer caused by extra rewards in Bitcoin and Ethereum will no longer exist.

3.1.3 Forging Process

No matter which kind of forks appearing in the chain, it will be the correct mainchain if it is with the highest voting right. Since the main forger has extremely high voting right, it will reach a consensus within a short block length. To facilitate understanding, we just describe the forging process and selection strategy of the top 2 highest voting right forgers. Figure 1 describes the act of a forging group in ideal environment. R_n in the figure below represents the n-th node for voting right in this forging committee group.

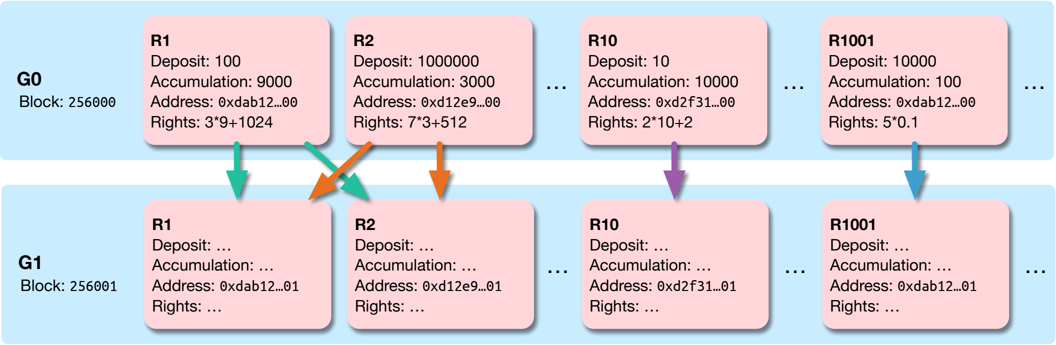

Figure 1: Act of a Forging Group in Ideal Environment.

Since the network environment in actual world is complicated, broadcast sent by the voting committee member with the highest voting right may not be accepted by the next forging group. Figure 2 shows the alternative scheme in such a case.

The total voting right relates to grouping, cash deposit, accumulated block height, ranked voting right and address, so it is hard for committee members to conspire to cheat. But network issues or other incidental reasons may lead the forger with the highest voting right to accept the block created by the forger of the second highest voting right in the previous block. Based on CPoS mechanism, the fork can always be removed within a short block height, as shown in figure 3. There are forks at block height 256,000 and 256,001. And the

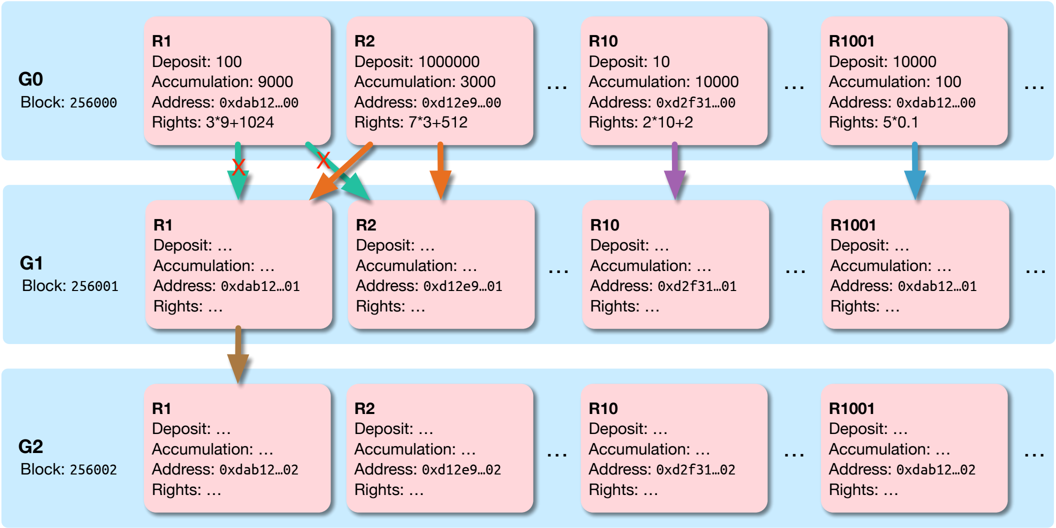

Figure 2: Act of a Forging Group When Broadcast is not Accepted.

forger with the highest voting right in G_2 group chooses one chain in it. The total voting right becomes remarkably greater and has a relatively high probability to win. Forger in G_3 group will continue to create blocks based on this chain.

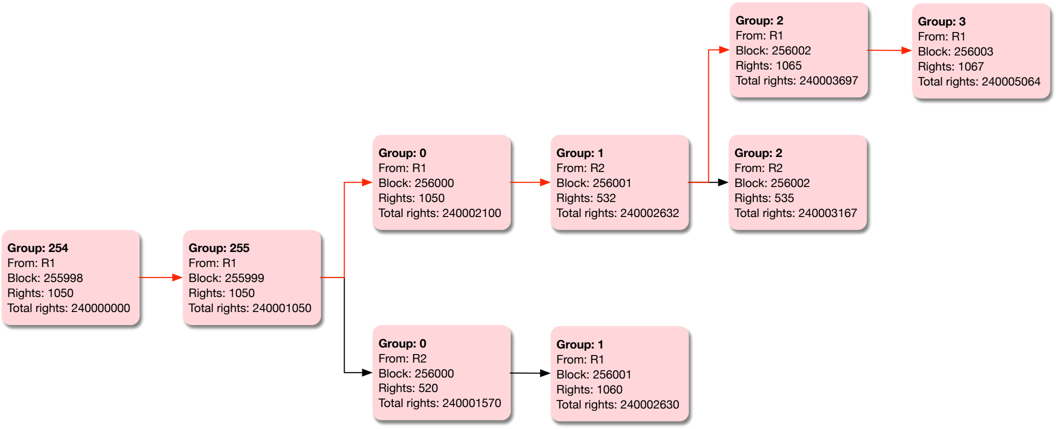

Figure 3: Fork can Always be Removed Within a Relatively Short Block Height.

3.1.4 Double Spending Attack

If a user initiates a transaction first and then makes a contradictory second transaction before the first one was confirmed, which means the first transaction has not been packed into a block at this moment. Then, the fraudsters deliberately broadcast the first transaction to half of the network, and broadcast the second one to the other half of the network respectively. Coincidentally, two miners on both side of the network get the right for bookkeeping at the same time, construct their blocks respectively and publish their block to everyone in the network. At this time, the original unified ledger is forked.

The fraudsters will take the following steps to maximize their illegal benefits. If one of the branches is approved, which means the corresponding transaction is confirmed by blockchain and the fraudsters get the goods ordered by the cryptocurrency, they immediately becomes miners and take part in the competition for the next right of bookkeeping in another branch. If they construct a new block on the second branch, and make the total rights of second branch higher than the first one, the second branch will be approved as the main chain while the first branch will be abandoned and the transaction in this branch will no longer be valid. But the goods of this transaction has already been delivered to the fraudsters. Consequently, the double spending attack is successful.

To prevent this attack, the rules of the forging committee should be reviewed to enhance the system. The voting rights of the forging members are related to the value of the deposit. The initial voting right of a member can be expressed by the following formula:

$$K = p[log_{10}(d) - 2] \tag{1}$$

Where d represents the amount of deposit and p is a random value between 1 to 4 which is generated when the forging member first obtains the voting rights and will be recalculated when the voting rights is initialized again. A newly applied forging member does not get the voting rights immediately, and will need to wait for 256,000 blocks before gaining the voting rights. After that, the voting rights of this member increases K after every 256 block height (i.e., one round). However, To reduce the risk of joint attacks by malicious members, the voting rights will stop increasing after 2,560 block height. Thus, the total rights can be evaluated by the following formula:

$$T = p[log_{10}(d) - 2] \times acc \tag{2}$$

Where acc represents the accumulation of block heights by rounds whose maximum is 10. Forging members are then ranked by their voting rights T, and the top 10 members in one group will get an additional ranking voting rights R, which can be evaluated by:

$$R = 2^{11-r} \tag{3}$$

Where r represents the rank of forging member. Thus, the total voting rights of the top 10 members can be expressed by:

$$T_r = p[log_{10}(d) - 2] \times acc + 2^{11-r} \tag{4}$$

To perform double spending attack, a fraudster should make the total rights of the block ranked second in total voting rights exceed the first one after the blockchain fork. To

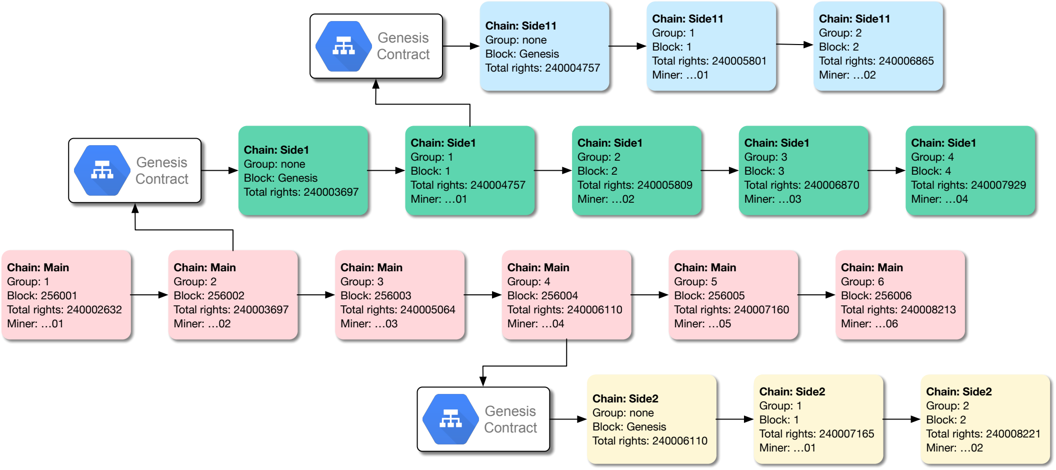

Figure 4: Super Sidechain Tree.

achieve thid goal, the fraudster should make his T more than the difference of additional rights between the first and second ranked member, which is $R_1 - R_2$ in equation 3. Thus, we can get the following inequality:

$$p[log_{10}(d) - 2] > R_1 - R_2 = 512 \tag{5}$$

Solving the inequality, we can get that when the random value p gets its maximum value of 4, the deposit d should be no less than $10^{14.8}$, while it will be no less than $10^{53.2}$ when p gets its minimum value of 1. Thus, to achieve this attack, one should pay at least $10^{14.8}$ as the deposit, which is almost impossible. Consequently, double spending attack is extremely difficult to implement in GaiaWorld system.

3.2 Super Sidechain

The code of super sidechain is same as that of mainchain with the same consensus algorithm (CPoS). The super sidechain can directly use part of nodes network in the mainchain, or have its own deployed node network.

3.2.1 Sidechain Tree

Super sidechain supports to take designated sidechain as the mainchain and continue to fork to new sidechains to generate a sidechain tree, as shown in figure 4. In each chain, either smart contract or transaction can be a complete business. Since sidechains and mainchain are in different node network, each blockchain has different generator and rule. Thus we formulate a cross-chain business specification to guarantee that sidechain and mainchain can run a complete business.

3.2.2 Independence and Flexibility of Sidechain

The advantage of sidechain architecture is data independence, no extra communication and storage burden to mainchain, no excessive data expansion or broadcasting, and no code bug spreading to other chains. And sidechains provide users with an optional flexibility. Parameters of all blockchains can be customized, such as block interval, block rewards, the use of transaction fee etc. Besides, since the business logic can be customized, new type of transactions and smart contracts can be developed based on sidechains.

The developer of sidechain does not need to provide assets and only the nodes owner shall have this currency, since the existing mainchain nodes can be used. Thus developers does not need to consider the problem of transaction platform. The sidechain can be release automatically as long as the smart contract is compiled according to the sidechain publication standard. Thus, developers only need to consider the specific business logic.

3.3 Security Function

Security function is a specific smart contract according to security standard in the system. This smart contract can be executed in smart contract virtual machine and communicate with external server to collect data or finish a complete business.

3.3.1 Closure and Extensibility

Standard smart contract is essentially codes running in virtual machine, all data operated by which are stored in the chain. Under such an architecture, smart contract code cannot call and operate data outside the blockchain. This results in that traditional smart contract cannot interact with data outside of blockchain. In GaiaWorld blockchain architecture, we provide a secure API for external call which can be customized by users. Thus, codes running in GaiaWorld chain can take advantage of security function to call external data, which will greatly expand application scope of smart contract.

3.3.2 Security Consensus

If a function is not regarded to be secure, the miner will reject to execute the smart contract containing this function. Therefore, security consensus should be addressed beforehand. We introduce 3 rules for security definition and protection.

First, smart contract which provide security function should pay cash deposit, which would be fined for fraud. Then, forging committee should vote for the smart contract before it can be executed. If the majority of the committee votes in favor, the security function can be called afterwards. Last, a vote of no confidence can be held at any time, and the deposit will be forfeited once more than half of the vote is taken. Thus, security function shall try best to persuade committee to believe their security by providing source code or any other authoritative evidentiary materials.

3.3.3 Distributed Transaction Standard

In a complicated and consistent application scenario, such as multi-party transactions, a simple security function cannot meet the requirements. Therefore, standard of distributed transaction of security functions is defined to guarantee that the blockchain can conduct secure and consistent data exchange with external data.

4 Evaluation

A prototype of Gaia was implemented using the rust language. We used 50 ordinary commercial personal computers, each of which started up 1-30 different nodes, and each of node could use 1-100 different addresses at the same time, to simulate the situation of up to 1,500 nodes and 150,000 addresses. We allocated 20Mbps of Intranet bandwidth to each node. In order to simulate the real network environment, we set 200ms of information transmission delay for each node, and each node could connect up to 125 other nodes. The default block generation speed we set in the Genesis Block is 1000ms.

4.1 The Influence of the Number of Nodes on the Block Confirmation Speed

We simulated the average time from transactions issued until the first confirmations obtained, and the average generation speed of blocks, with each node using 2 block addresses. As shown in Figure 5, the generation speed of blocks does not change significantly as the number of nodes increases. However, the confirmation speed of the blocks increases significantly as the number of nodes increases, and the upward trend is approximately linear.

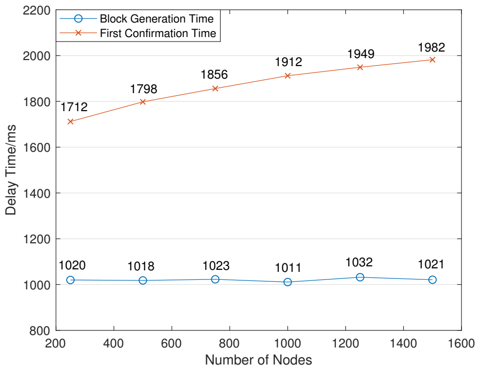

Figure 5: The Influence of the Number of Nodes on the Block Confirmation Speed.

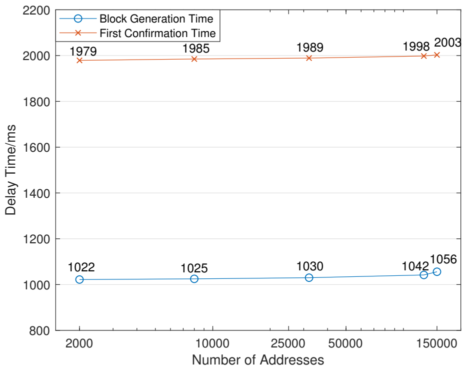

Figure 6: The Influence of the Number of Addresses on the Block Confirmation Speed.

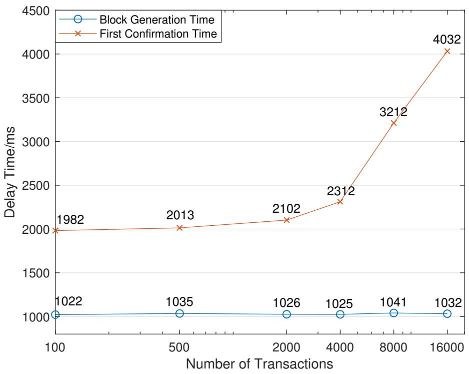

Figure 7: The Influence of the Number of Transactions on the Block Confirmation Speed.

4.2 The Influence of the Number of Addresses on the Block Confirmation Speed

We simulated the average time from transactions issued until the first confirmations obtained, and the average generation speed of blocks, with 1500 nodes using different block addresses. As shown in Figure 6, the generation speed and the confirmation speed of blocks remains almost unchanged as the number of addresses increases, which means that the number of addresses used by each node does not substantially affect the overall performance of the system.

4.3 The Influence of the Number of Transactions on the Block Confirmation Speed

We simulated the average time from transactions issued until the first confirmations obtained, and the average generation speed of blocks, with 1500 nodes using 150,000 addresses and different number of transactions. As shown in Figure 7, the generation speed of blocks does not change significantly when the number of transactions increases, but the confirmation time of blocks increases significantly as the number of transactions increases.

5 Application Scenarios of GaiaWorld

As a new Internet underlying architecture, functions to be realized by GaiaWorld are far more than digital currency or electronic contracts. Thanks to new technology brought by GaiaWorld, we are capable to reconstruct most of Internet applications in a decentralized manner.

5.1 High Performance Requirement Applications

Modern Internet applications have an increasingly high requirement for response speed and server processing capability. In view of a huge amount of data, even to integrate all existing blockchain network cannot meet the demand since the new increased computing resources are mostly wasted in validation for computing power competition and to avoid unexpected fork.

GaiaWorld based on CPoS mechanism can greatly improve the efficiency of blockchain. In small scale network, we have already realized 2,000 times/second processing efficiency. And since nodes in CPoS blockchain do not need computing power competition for right charging, performance of the whole network can be further improved with the expansion of network scope.

5.2 Interact with Applications Outside the Blockchain

Traditional blockchain applications generally refers to a certain kind of smart contract to conduct simple conditional judgement and automatically handle transactions. The problem lies in that these smart contracts cannot access data outside of blockchain, which limits the functionality of these applications.

In GaiaWorld, developers can use customized security function to solve this problem. A set of safe and anti-tampering security functions can be deployed for developers to gain data outside the blockchain through gateway. This process is bi-directional, which means that smart contract can also send data to addresses outside by security functions.

5.3 Sidechain Applications

The complexity of a blockchain network depends on the quantity of asset and application running in the chain. As to some blockchain applications which are complicated and not frequently interacting with blockchain resources in core function such as currency and authentication, to run directly in mainchain in the form of smart contract is not a good choice.

Therefore, GaiaWorld provides super sidechain for application. Since super sidechain supports tree-like and multi-layer sidechain generation, developers can derive sidechain recursively. Besides, super sidechain can provide developers with safe and isolable application development. And if complicated application runs in independent super sidechain, it can not only improve the execution efficiency of improving application itself, but also greatly reduce the bloated degree of mainchain.

5.4 Time Blockchain Application

Traditional blockchain keeps all blocks since the beginning, but GaiaWorld sidechain can support blocks which only keep a certain time length. This can effectively reduce the length of blockchain and storage pressure, which enables GaiaWorld to be deployed to most equipment with relatively low performance.

6 Conclusions

In this paper, we introduce GaiaWorld, a new blockchain system which is designed for decentralized applications. GaiaWorld introduces a new consensus mechanism called CPoS, which can reduce resource consumption and promote degree of decentralization compared with PoW and traditional PoS mechanisms. By introducing the mechanism of forging committee and forging group, GaiaWorld can quickly remove the forks on the premise of ensuring decentralization. In addition, the super sidechain and security function provides developers with chances to build secure and isolated developing environment, and help them build and manage complex on-chain applications.

Acknowledgement: The authors wish to express their appreciation to the reviewers for their helpful suggestions which greatly improved the presentation of this paper.

References

Aitzhan, N. Z.; Svetinovic, D. (2016): Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams. *IEEE Transactions on Dependable and Secure Computing*.

Bentov, I.; Lee, C.; Mizrahi, A. et al. (2014): Proof of activity: Extending bitcoin's proof of work via proof of stake. *ACM SIGMETRICS Performance Evaluation Review*, vol. 42, no. 3, pp. 34–37.

Biryukov, A.; Pustogarov, I. (2015): Proof-of-work as anonymous micropayment: Rewarding a tor relay. In *International Conference on Financial Cryptography and Data Security*, pp. 445–455. Springer.

Buterin, V. (2013): Ethereum white paper, 2014. *URL https://github.com/ethereum/wiki/wiki/White-Paper*.

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

Chen, T.; Li, X.; Wang, Y. et al. (2017): An adaptive gas cost mechanism for ethereum to defend against under-priced dos attacks. In *International Conference on Information Security Practice and Experience*, pp. 3–24. Springer.

Chen, Z.; Zhu, Y. (2017): personal archive service system using blockchain technology: case study, promising and challenging. In *AI & Mobile Services (AIMS), 2017 IEEE International Conference on*, pp. 93–99. IEEE.

Christidis, K.; Devetsikiotis, M. (2016): Blockchains and smart contracts for the internet of things. *Ieee Access*, vol. 4, pp. 2292–2303.

Gervais, A.; Karame, G. O.; Wüst, K. et al. (2016): On the security and performance of proof of work blockchains. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3–16. ACM.

Hirai, Y. (2017): Defining the ethereum virtual machine for interactive theorem provers. In *International Conference on Financial Cryptography and Data Security*, pp. 520–535. Springer.

Huh, S.; Cho, S.; Kim, S. (2017): Managing iot devices using blockchain platform. In *Advanced Communication Technology (ICACT), 2017 19th International Conference on*, pp. 464–467. IEEE.

Jentzsch, C. (2016): Decentralized autonomous organization to automate governance. *White paper, November*.

Kim, T. (2017): On the transaction cost of bitcoin. *Finance Research Letters*, vol. 23, pp. 300–305.

King, S.; Nadal, S. (2012): Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. *self-published paper, August*, vol. 19.

Lin, G. (2018): Phishing detection with image retrieval based on improved texton correlation descriptor. *Computers, Materials & Continua*, vol. 57, no. 3, pp. 533–547.

Nakamoto, S. (2008): Bitcoin: A peer-to-peer electronic cash system.

Norta, A. (2015): Creation of smart-contracting collaborations for decentralized autonomous organizations. In *International Conference on Business Informatics Research*, pp. 3–17. Springer.

Omohundro, S. (2014): Cryptocurrencies, smart contracts, and artificial intelligence. *AI matters*, vol. 1, no. 2, pp. 19–21.

Pilkington, M. (2016): 11 blockchain technology: principles and applications. *Research handbook on digital transformations*, pg. 225.

Ron, D.; Shamir, A. (2013): Quantitative analysis of the full bitcoin transaction graph. In *International Conference on Financial Cryptography and Data Security*, pp. 6–24. Springer.

Saito, K.; Yamada, H. (2016): What's so different about blockchain?-blockchain is a probabilistic state machine. In *Distributed Computing Systems Workshops (ICDCSW), 2016 IEEE 36th International Conference on*, pp. 168–175. IEEE.

Sergey, I.; Hobor, A. (2017): A concurrent perspective on smart contracts. In *International Conference on Financial Cryptography and Data Security*, pp. 478–493. Springer.

Shi, N. (2016): A new proof-of-work mechanism for bitcoin. *Financial Innovation*, vol. 2, no. 1, pp. 31.

Swan, M. (2015): Blockchain thinking: The brain as a dao (decentralized autonomous organization). In *Texas Bitcoin Conference*, pp. 27–29.

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

Watanabe, H.; Fujimura, S.; Nakadaira, A. et al. (2015): Blockchain contract: A complete consensus using blockchain. In *Consumer Electronics (GCCE), 2015 IEEE 4th Global Conference on*, pp. 577–578. IEEE.

Watanabe, H.; Fujimura, S.; Nakadaira, A. et al. (2016): Blockchain contract: Securing a blockchain applied to smart contracts. In *Consumer Electronics (ICCE), 2016 IEEE International Conference on*, pp. 467–468. IEEE.

Wood, G. (2014): Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, vol. 151, pp. 1–32.

Xu, W.; Xiang, S.; Sachnev, V. (2018): A cryptograph domain image retrieval method based on paillier homomorphic block encryption.

Xu, X.; Pautasso, C.; Zhu, L. et al. (2016): The blockchain as a software connector. In *2016 13th Working IEEE/IFIP Conference on Software Architecture (WICSA)*, pp. 182–191. IEEE.

Zhang, F.; Cecchetti, E.; Croman, K. et al. (2016): Town crier: An authenticated data feed for smart contracts. In *Proceedings of the 2016 aCM sIGSAC conference on computer and communications security*, pp. 270–282. ACM.

Zheng, Z.; Xie, S.; Dai, H. et al. (2017): An overview of blockchain technology: Architecture, consensus, and future trends. In *Big Data (BigData Congress), 2017 IEEE International Congress on*, pp. 557–564. IEEE.

Zhu, Y.; Guo, R.; Gan, G. et al. (2016): Interactive incontestable signature for transactions confirmation in bitcoin blockchain. In *Computer Software and Applications Conference (COMPSAC), 2016 IEEE 40th Annual*, volume 1, pp. 443–448. IEEE.

Zikratov, I.; Kuzmin, A.; Akimenko, V. et al. (2017): Ensuring data integrity using blockchain technology. In *Open Innovations Association (FRUCT), 2017 20th Conference of*, pp. 534–539. IEEE.