Digital Communications and Networks 6 (2020) 480–485

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

Performance analysis and comparison of PoW, PoS and DAG based blockchains

Bin Cao [a,b], Zhenghui Zhang [c], Daquan Feng [d,*], Shengli Zhang [e], Lei Zhang [f], Mugen Peng [a], Yun Li [c]

[a] The State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China
[b] The State Key Laboratory of Integrated Services Networks (Xidian University), Xian, 710000, China
[c] School of Communications and Information Engineering and Chongqing Key Lab of Mobile Communications Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
[d] The Guangdong Key Laboratory of Intelligent Information Processing, College of Electronic and Information Engineering, Shenzhen University, Shenzhen, 518060, China
[e] College of Information Engineering, Shenzhen University, Shenzhen, 518060, China
[f] School of Engineering, University of Glasgow, Glasgow, G12 8QQ, UK

ARTICLE INFO

Keywords:
Blockchain
Proof of work
Proof of stake
Direct acyclic graph
Performance comparison

ABSTRACT

In the blockchain, the consensus mechanism plays a key role in maintaining the security and legitimation of contents recorded in the blocks. Various blockchain consensus mechanisms have been proposed. However, there is no technical analysis and comparison as a guideline to determine which type of consensus mechanism should be adopted in a specific scenario/application. To this end, this work investigates three mainstream consensus mechanisms in the blockchain, namely, Proof of Work (PoW), Proof of Stake (PoS), and Direct Acyclic Graph (DAG), and identifies their performances in terms of the average time to generate a new block, the confirmation delay, the Transaction Per Second (TPS) and the confirmation failure probability. The results show that the consensus process is affected by both network resource (computation power/coin age, buffer size) and network load conditions. In addition, it shows that PoW and PoS are more sensitive to the change of network resource while DAG is more sensitive to network load conditions.

1. Introduction

Recently, the blockchain, which builds a decentralized, shared network for secure and reliable data record and transfer without a centralized authority and is one of core technologies of the 5th Generation(5G) [1], has become a hot topic in business, industry and academia [2]. In the blockchain, the consensus mechanism makes the network reach an agreement in the presence of faults [3], and this is the key to building distributed trustworthiness among users. The blockchain is proposed as a foundational technology of Bitcoin, which is a peer-to-peer based distributed ledger for establishing trust. More recently, various blockchain consensus mechanisms have been proposed [4] and most of them are based on three mainstream mechanisms, namely, Proof of Work (PoW), Proof of Stake (PoS) and Direct Acyclic Graph (DAG) [5–7]. As the first blockchain, Bitcoin is the most famous project with the highest market value [8], which is based on PoW. PoS, and the future plan of Ethereum [9], has been proposed to address the limitation of PoW and adopted in various blockchain systems, such as Nxt. In order to improve the processing rate to face the exponential increase of transactions, the DAG-based blockchain is developed by IOTA and Byteball [7,10]. Since PoW, PoS and DAG are widely used and most proposed consensus mechanisms are based on them, it is typical and meaningful to choose these consensus mechanisms for performance analysis and comparison.

PoW is the most classical consensus mechanism in the blockchain that is pioneered by Bitcoin. Its core idea is that members of the system (miner) use their computing power to compete the hashing operation (SHA-256) [11]. The winner who first finds the hash value lower than the announced target has the right to insert a new block into the blockchain and get a certain amount of reward. Different from PoW that consumes a lot of computing resources, PoS proposes a conception of coin age, which is unspent asset multiplied by its duration from the last winning time to the current time, to avoid high resource consumption of the competition process. In PoS, the consensus process also relies on the hashing

* Corresponding author.
E-mail address: fdquan@szu.edu.cn (D. Feng).

https://doi.org/10.1016/j.dcan.2019.12.001
Received 27 May 2019; Received in revised form 30 August 2019; Accepted 23 December 2019
Available online 3 January 2020

operation, and a higher coin age will lead to a higher probability for the node to win the right of creating a new block. A typical PoS-based blockchain is Nxt [6].

Unlike PoW and PoS, there is no competition to create a new block in DAG, and all transactions are connected directly or indirectly [7,10,12]. As the first DAG-based blockchain, Tangle [7] is proposed for the micro-transaction in the Internet of Things (IoT) [13,14]. As shown in Fig. 1, it allows different branches to eventually merge into the chain. In Tangle, any new arrival transactions could access the blockchain network immediately as a new vertex (or tip) as long as they approve a number of unconfirmed transactions (typically two with the random selection). The confirmation is achieved when the cumulative weight (which is the sum of its own weight and the weights of other transactions) reaches the predefined threshold. As a result, the confirmation rate and Transaction Per Second (TPS) would be much higher than PoW and PoS when the new transaction arrival rate is fast.

Although PoW, PoS and DAG have been widely used in blockchains and revealed some advantages and benefits, there still remain some challenges and limits: (1) For PoW, the high computing difficulty would cause deteriorated and meaningless energy consumption. (2) For PoS, it is beneficial for the wealthy miner, and might cause oligopolies or near-monopolies. (3) For DAG, since the new transaction may not arrive quickly and steadily all the time in practical IoT scenarios, the consensus delay would be deteriorated when the traffic load becomes low.

As is well known, the above-mentioned issues are very crucial in the blockchain applications, especially in IoT scenarios where the computational capacity and energy are limited and the network load is randomly generated. However, there lacks an analysis to provide an insightful understanding of these mechanisms. To this end, this work investigates the performance of PoW, PoS and DAG based blockchains with mathematical analysis.

2. System model and definitions

For a fair comparison and discussion, we assume that there are n nodes running the PoW, PoS or DAG based blockchain in the system, where all nodes are connected directly in a single-hop network. We ignore the transmission delay in our analysis since it is much shorter than the computation time and new transaction arrival interval. We also assume that the maximum number of new transactions in a block in PoW and PoS is L, and the queueing length at node i is Q_i. In Tangle, the new transaction has to approve the previous transactions as soon as possible. Thus, any vertex in Tangle can be viewed as a block that only records one transaction; it is also called as a transaction instead of a block sometimes [7]. Therefore, we set $L = 1$ in DAG. In addition, we consider that the new transaction follows a Poisson distribution with the arrival rate λ_i at node i, and that the weight of any transaction is 1 in DAG.

In this work, we use three typical blockchains as examples, namely, Bitcoin (PoW), Nxt (PoS) and Tangle (DAG), to analyze the performance of the three consensus mechanisms. In Bitcoin and Nxt, when the cumulative blocks reach a predefined number (it is six in Bitcoin, and ten in Nxt), the transaction could be confirmed. In Tangle, the cumulative weight is used to indicate the consensus process level, and it would increase gradually as a result of the approval of new transactions through random selection.

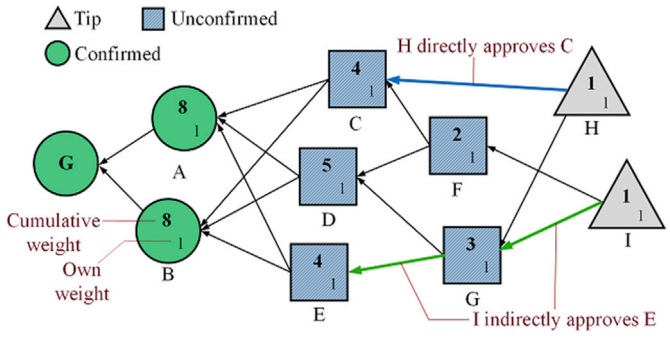

Fig. 1. A typical example of Tangle.

We introduce four metrics to evaluate the performance of the blockchain system. In order to show the performance of the consensus mechanism in the blockchain system, we define the average time to generate a new block, the confirmation delay and TPS to demonstrate effectiveness, and the confirmation failure probability to illustrate robustness as follows. The average time is the time consumption to generate a new block to show the block processing rate, the confirmation delay is to show the processing rate from the new transaction arrival to the final confirmation, the TPS is the transaction processing capacity to show the throughput in the blockchain, and the confirmation failure probability is to show QoS in the consensus process.

3. Performance analysis

In this section, we present an analysis and comparison of the PoW, PoS and DAG based blockchains in terms of blocking efficiency, TPS, confirmed probability and confirmation delay to provide an easy-understanding for the three main consensus mechanisms.

3.1. PoW

According to Ref. [15], all legal blocks must satisfy the following computing condition:

$$U \le \frac{1}{D} \le 1 \tag{1}$$

where U is the value obtained by hash operation and D is the target difficulty. To win the hash operation competition, a miner must try its best (using all of the computational power) to find a U to satisfy the above condition.

Based on [15], to find a feasible U, we can first obtain the mining time for node i to generate U in PoW (T_{m_i}), which is a function of $\frac{D}{r_i}$, where r_i is the computational power at node i for mining in PoW and the cumulative probability distribution can be expressed as

$$P\{T_{m_i} \le t\} = 1 - \exp\left[\log\left(1 - \frac{1}{D}\right) r_i t\right] \tag{2}$$

In a practical system, $\frac{1}{D} << 1$. In this case, $\log\left(1 - \frac{1}{D}\right) \approx -\frac{1}{D}$ and thus $P\{T_{m_i} \le t\} \approx 1 - \exp\left(-\frac{r_i}{D} t\right)$. Moreover, we can see that T_{m_i} follows an exponential distribution with mean $\frac{r_i}{D}$, and the expected mining time for miner i is $E[T_{m_i}] = \frac{D}{r_i}$.

Therefore, with n miners, assume that each miner has $r_1, r_2, \ldots, r_n$ with the corresponding mining time $T_{m_1}, T_{m_2}, \ldots, T_{m_n}$, the average time to generate a new block in PoW (T_b^{pow}) is the minimum time in them. According to the properties of the exponential distribution, we can have

$$T_b^{pow} \overset{def}{=} \min\{T_{m_1}, T_{m_2}, \ldots, T_{m_n}\} = \frac{D}{\sum_{i=1}^{n} r_i} \tag{3}$$

PoW requires to wait for a number of cumulative blocks to confirm (m^{pow}). Thus, the confirmation delay can be written as

$$T_c^{pow} = m^{pow} \times T_b^{pow} \tag{4}$$

Due to the limitation of block size, no matter how many new transactions arrive in the duration of T_b^{pow}, the maximum number of processed transactions cannot exceed the maximum blocksize, L, which is between two consecutive blocks. Therefore, the TPS is determined by the block size limitation L and the new transaction arrival rate λ, simultaneously, it can be calculated as follows:

$$TPS^{pow}=\begin{cases}\sum_{i=1}^{n}\lambda_i, & \sum_{i=1}^{n}\lambda_i \le \frac{L}{T_b^{pow}}\\ \frac{L}{T_b^{pow}}, & \sum_{i=1}^{n}\lambda_i > \frac{L}{T_b^{pow}}\end{cases} \tag{5}$$

when $\sum_{i=1}^{n}\lambda_i \le \frac{L}{T_b^{pow}}$, it implies that the network load is light, and thus all new arrival transactions could be recorded in the block generation period. Otherwise, it is the heavy load scenario, $\sum_{i=1}^{n}\lambda_i > \frac{L}{T_b^{pow}}$. In this case, the new block cannot record all of the new coming transactions anymore.

In PoW (similar as in PoS), in the case of winning to broadcast the new block, the miner should record all of the new transactions and finally all miners cached the same information. Assume $Q_i \ge L$, when $\sum_{i=1}^{n}\lambda_i \le \frac{L}{T_b}$, the network load is light, and all new transactions could be fully recorded in a block period. In contrast, when $\sum_{i=1}^{n}\lambda_i > \frac{L}{T_b}$, the network becomes heavily loaded and only L transactions can be processed in T_b. Thus, the rest should wait in queue. If the queue is full, the new coming transaction would be discarded and the failure with happen (here we use FIFO).

Consequently, the confirmation failure probability of PoW (P_f^{pow}) can be calculated as follows:

$$P_f^{pow}=\begin{cases}0, & \sum_{i=1}^{n}\lambda_i \le \frac{L}{T_b^{pow}}\\ 1-\frac{L}{\sum_{i=1}^{n}\lambda_i T_b^{pow}}, & \sum_{i=1}^{n}\lambda_i > \frac{L}{T_b^{pow}}\end{cases} \tag{6}$$

3.2. PoS

Similar with PoW, in PoS, all valid blocks also need to satisfy the condition $U \le \frac{bal_i \times t_i}{D} \le 1$, where $bal_i \times t_i$ is the coin age of miner i, bal_i denotes the balance (or say stake) of miner i in the system, t_i is the lifetime of bal_i from the last winning time to the current time and it will be reset to zero when the miner wins. Similar with the analysis in PoW, the expected minimum mining time with n miners to generate a new block in PoS (T_b^{pos}) is

$$T_b^{pos}=\frac{D}{\sum_{i=1}^{n} bal_i \times t_i} \tag{7}$$

Since it needs waiting m^{pos} blocks to confirm the transaction, the confirmation delay in PoS (T_c^{pos}) can be expressed as

$$T_c^{pos}=m^{pos}\times T_b^{pos} \tag{8}$$

Consequently, the TPS and the confirmation failure probability in PoS (similar with PoW) (TPS^{pos} and P_f^{pos}) can be calculated as follows:

$$TPS^{pos}=\begin{cases}\sum_{i=1}^{n}\lambda_i, & \sum_{i=1}^{n}\lambda_i \le \frac{L}{T_b^{pos}}\\ \frac{L}{T_b^{pos}}, & \sum_{i=1}^{n}\lambda_i > \frac{L}{T_b^{pos}}\end{cases} \tag{9}$$

$$P_f^{pos}=\begin{cases}0, & \sum_{i=1}^{n}\lambda_i \le \frac{L}{T_b^{pos}}\\ 1-\frac{L}{\sum_{i=1}^{n}\lambda_i T_b^{pos}}, & \sum_{i=1}^{n}\lambda_i > \frac{L}{T_b^{pos}}\end{cases} \tag{10}$$

3.3. DAG

Different from PoW and PoS, the DAG-based blockchain allows the new transaction to perform consensus without the assistance of miner. A typical DAG-based blockchain is Tangle [16], which aims to address the huge micro-transactions in IoT systems. In Tangle, the new transaction has to approve the previous transactions as soon as possible. Thus, any vertex in Tangle can be viewed as a block that only records one transaction. As a result, T_b^{dag} is determined by the new transaction arrival rate, which is shown as follows:

$$T_b^{dag}=\frac{1}{\sum_{i=1}^{n}\lambda_i} \tag{11}$$

Define h_r as the revealing time to update the transaction and consensus information in the blockchain network, which is discussed in Ref. [7]. When $\sum_{i=1}^{n}\lambda_i > \frac{1}{h_r}$, the network is heavily loaded. In contrast, when $\sum_{i=1}^{n}\lambda_i \le \frac{1}{h_r}$, the network is lightly loaded. According to the previous analysis in Ref. [7], we can know the growth rate of the cumulative weight, $H(t)$, which can be expressed as follows:

$$H\left(t\right)\begin{cases}1+\sum_{i=1}^{n}\lambda_i, & t\in\left(0,\infty\right), \sum_{i=1}^{n}\lambda_i \le \frac{1}{h_r}\\ 2\exp\left(\frac{0.352t}{h_r}\right), & t\in\left(0,t_0\right), \sum_{i=1}^{n}\lambda_i > \frac{1}{h_r}\\ 2\exp\left(\frac{0.352t_0}{h_r}\right)+\sum_{i=1}^{n}\lambda_i\left(t-t_0\right), & t\in\left(t_0,\infty\right), \sum_{i=1}^{n}\lambda_i > \frac{1}{h_r}\end{cases} \tag{12}$$

$H(t)$ increases linearly with the overall arrival rate in light load, while in the heavy load it first increases exponentially during the adaptation time ($t_0 \approx 2.84 h_r \ln(2\sum_{i=1}^{n}\lambda_i h_r)$),[1] and then increases linearly when the adaptation time ends.

Therefore, considering the weight threshold for confirmation, the confirmation delay in DAG (T_c^{dag}) can be expressed as follows:

$$T_c^{dag}=\frac{W}{H(t)} \tag{13}$$

where W is the cumulative weight threshold for confirmation in DAG. Since the TPS increases with the new transaction arrival rate without limitation in DAG, the TPS in DAG (TPS^{dag}) can be shown as follows:

$$TPS^{dag}=\sum_{i=1}^{n}\lambda_i \tag{14}$$

Moreover, because any new transaction can access the DAG-based blockchain as soon as possible in a parallel and distributed manner, there is no transaction lost which may happen in PoW and PoS. As a result, confirmation failure probability in DAG, P_f^{dag} is zero.

4. Simulation and discussions

In this section, we evaluate the performance of PoW, PoS and DAG in terms of the average time to generate a new block, the confirmation delay, the TPS and the confirmation failure probability. Without loss of generality, we assume that the blockchain network includes ten miners/nodes ($n = 10$). For a fair comparisons, let each miner in PoW and PoS (with $t_i = 1$ second) have equal computational capacity on the hash

[1] For more details, please refer to Ref. [5].

operation, and assume that $r_i = 1$ TH/second in PoW and $bal_i = 10^{12}$ Nxt in PoS. According to the settings in the real system, we also assume that $D = 10^{13}$ [8] and $L = 255$ [6], respectively. The revealing time in DAG, $h_r = 1$ second, and the weight threshold for confirmation, $W = 200$. The weight of each transaction is 1, and thus the cumulative weight increases by 1 gradually.

4.1. The average time to generate a new block

First, we show the Cumulative Distribution Function (CDF) of T_b with $\lambda_i = 1$ in Fig. 2. We can see that CDF of DAG increases most quickly, the reason is that DAG lets the new transaction attach to the blockchain network as soon as it arrives. The CDF of PoW is higher than that of PoS when the time is less than 1 s. In contrast, when the time is more than 1 s, the CDF of PoS becomes higher than that of PoW. This is because the coin age increases over time, and thus the probability to compute the value for hash operation increases.

Next, the average time, T_b, in PoW, PoS and DAG, which are affected by the computational power, coin age and new transaction arrival rate respectively, are shown in Fig. 3. Obviously, with the increase of resources to achieve a consensus, the average time to generate a new block becomes shorter. Since PoW needs a lot of computational power, which may be impractical for the light nodes, such as the mobile equipment and IoT users. Therefore, PoS, which has the same procedure but does not require high computational power, might be a better option. Different from PoW and PoS, there is no miner in DAG, and the new transaction must perform a consensus to confirm earlier transactions. Thus, the faster the arrival rate, the smaller T_b could be achieved. To this end, DAG is more suitable for the scenario where transactions come frequently. Note that the average time to generate a new block in DAG is significantly faster than that in PoW and PoS, but the block size in DAG (a block only stores one transaction) is much smaller than that in PoW and PoS (hundreds of transactions are stored in a block).

4.2. Confirmation delay

Fig. 4 illustrates the confirmation delay in PoW, PoS and DAG. It can be seen that, with the increase of resources in terms of computational power, balance and new transaction, the consensus could be achieved more quickly. Due to different requirements for conformation (the typical value in Bitcoin, $m^{pow} = 6$, the typical value in Nxt, $m^{pos} = 10$, and $W = 200$ in Tangle), and the confirmation delay is also quite different. In DAG, the transaction is only confirmed when its cumulative weight reaches the threshold W. Thus, the corresponding confirmation delay in DAG is determined by the new transaction arrival rate and the value of W.

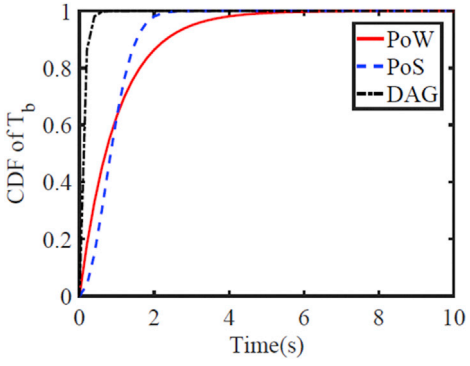

Fig. 2. CDF

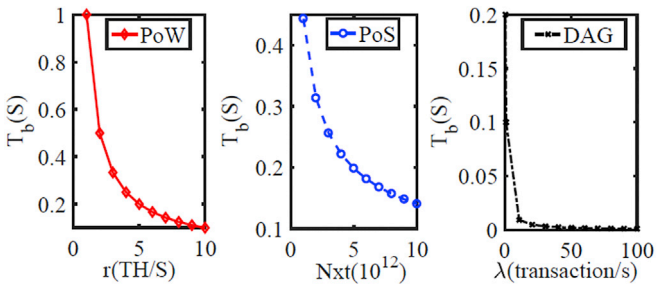

Fig. 3. The average time to generate a new block.

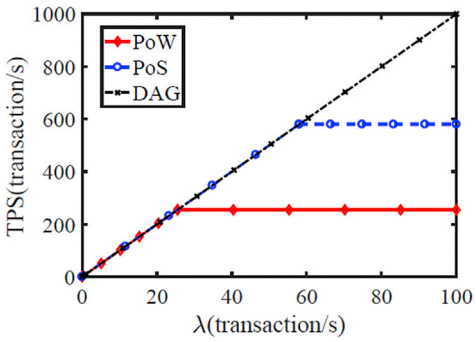

Fig. 4. Confirmation delay.

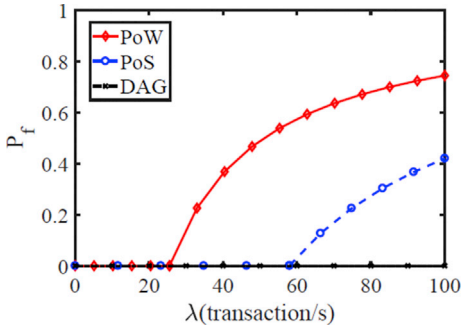

Fig. 5. TPS

4.3. TPS and confirmation failure probability

In order to show the impact of network load, we vary the new transaction arrival rate from the low load regime to the high load regime gradually. The TPS in PoW, PoS and DAG are illustrated in Fig. 5. It is shown that the TPSs in PoW and PoS increase linearly first until $TPS = \frac{L}{T_b}$, since they have achieved the limitation of block size. In contrast, the TPS in DAG always increases with the arrival rate without limitation. The result is consistent with the conclusion claimed in Ref. [7] that DAG-based blockchain uses the new transaction to perform consensus and its TPS has no technical upper bound as long as the new transaction arrives soon. Meanwhile, it is noted that the TPS of DAG would also decrease significantly if the new transaction arrives slowly, and even the consensus cannot be achieved with zero TPS in some extreme conditions. In Fig. 6, it is shown that for PoW and PoS, with the increase of the arrival rate, the network becomes heavily loaded and failure occurs, and the new arrival transaction has to be discarded because the node has limited queue buffer. In contrast, since any transaction would be a new vertex in DAG as long as it arrives, the failure cannot be incurred by the limitations of block size and queue. Therefore, the corresponding confirmation failure probability in DAG keeps zero all the time.

4.4. Discussions

According to the results of our previous simulations, we can know that PoW, PoS and DAG depend on the different resources, i.e., computational power, coin age and new transaction arrivals, to achieve a

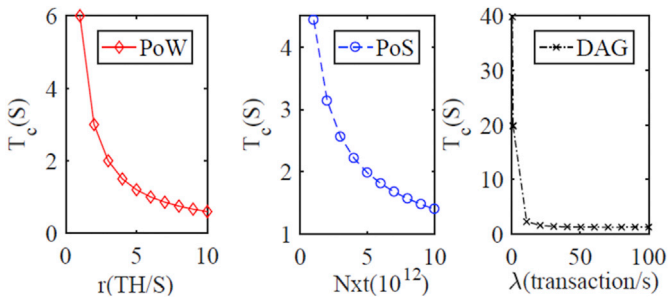

Fig. 6. Confirmation failure probability.

consensus. Generally speaking, PoW and PoS are competition-based mechanisms, and DAG is accumulation-orientated mechanism. PoW and PoS have a similar trend because both of them use the Hash algorithm to perform consensus process. In contrast, DAG motivates a new transaction to confirm the earlier ones instead of using heavy Hash computing. Moreover, in PoW and PoS, the single chain architecture is necessary to prevent forking, which limits the corresponding performance as well. As is well known, the latency can be 60 min (or 7 TPS) in Bitcoin and 3 min (or 20 to 30 TPS) in Ethereum, which could be too long to meet the exponential growth of IoT applications [17]. Due to the multi-chain architecture, the new transaction can be inserted in DAG as soon as possible. Technically, there is no upper bound of TPS. In summary, through these simulations, we can see clearly the performances and differences of various widely used consensus mechanisms, and the impact of key parameters on the consensus process.

5. Related works

As mentioned before, the consensus mechanism plays a key role in the performance of the blockchain. Except for PoW, PoS and DAG based blockchains, there are still some other consensus mechanisms, such as Proof of Authority (PoA), Proof of Burn (PoB), Practical Byzantine Fault Tolerance (PBFT) and Hash Graph [12,18–20].

PoA solves Byzantine fault errors by adding validators to ensure that the whole network reaches an agreement. Note that the validator in the network must have an active notary public license in the United States with no criminal records. PoB is a method to overcome the disadvantage of PoW, which utilizes the idea of burning coins to reduce the need for powerful computational resources when mining. Unlike the coin standing for the stake as a deposit in PoS, the coin in PoB will disappear forever once it is burned to be proof. PBFT is a new state-machine replication algorithm working in an asynchronous system, and is able to tolerate Byzantine faults with the polynomial complexity of $O(N^2)$. Hashgraph is a DAG-based blockchain. It relies on the gossip protocol and virtual vote mechanism to ensure that the algorithm can run efficiently and have high security in asynchronous environment.

In Ref. [21], the authors established a mathematical model to analyze the Tangle system and proved the existence of Nash equilibria for the system where a group of players tried to optimize their attachment strategies. In Ref. [22], the authors analyzed the block withholding attack in Bitcoins and proposed a strategy to deter the attack by blinding miners' ability. In Ref. [23], the authors developed a stochastic model for the evolution and dynamics of blockchain networks, where it provided a deeper understanding of crucial design issues for difficulty-of-work, block generation rate and adversarial attacks. In Ref. [24], the authors conceptualized a blockchain-based decentralized framework to design a crowdsourcing system, where the task of a requester can be solved by a crowd of workers without relying on any third-party trusted institution while guaranteeing the privacy and low transaction fee.

Although various consensus mechanisms, which show some improvements and benefits in the blockchain systems, have been proposed, there lacks an in-depth analysis to compare the performance of these consensus mechanisms. To this end, we present a performance analysis of three main blockchains in a mathematical manner, and conduct some experiments to provide a guideline to illustrate how the key parameters affect the performance of blockchains.

In this work, we compare and analyze the performance of blockchains based on PoW, PoS and DAG, respectively. As another important metric, security has been addressed a lot as well. Therefore, in the future, we plan to further analyze the security of blockchains to illustrate the differences and key factors of PoW, PoS and DAG. Meanwhile, as is well known, in order to achieve a consensus, resource allocation should be provided for communications. It is an interesting topic to investigate the resource allocation and communication overhead for consensus, and to study the differences between private blockchain (such as PBFT and Raft) and public blockchain (such as PoW and PoS) separately. Last but not least, as this work is only based on mathematical analysis and computational simulation, we are going to design a practical and realistic blockchain system for more experiments.

6. Conclusions

In this work, we analyze and discuss the consensus process in PoW, PoS and DAG based blockchains. We compare the main performance in terms of the average time to generate a new block, the confirmation delay, the TPS and the confirmation failure probability and illustrate the impact of computational power on PoW, balance on PoS and new arrival transactions on DAG, respectively. It is very important to know how to optimize the network performance by understanding different indexes. Through this work, it is easy to know some limitations of network parameters on the performance of blockchains, which can provide an understanding of the advantages and disadvantages using PoW, PoS and DAG to guide the implementation in a practical system.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61701059,Grant 61941114, and Grant 61831002, in part by the Fundamental Research Funds for the Central Universities of New TeachersProject, in part by the Chongqing Technological Innovation and Application Development Projects under Grant cstc2019jscx-msxm1322, and in part by the Eighteentg Open Foundation of State Key Lab of Integrated Services Networks of Xidian University under Grant ISN20-05.

References

[1] A. Morgado, K.M.S. Huq, S. Mumtaz, J. Rodriguez, A survey of 5G technologies: regulatory, standardization and industrial perspectives, Digital Commun. Netw. 4 (2) (2018) 87–97, https://www.sciencedirect.com/science/article/pii/S2352864817302584.
[2] M. Banerjee, J. Lee, K.R. Choo, A blockchain future for internet of things security: a position paper, Digital Commun. Netw. 4 (3) (2018) 149–160, https://www.sciencedirect.com/science/article/pii/S2352864817302900.
[3] M. Pease, R. Shostak, L. Lamport, Reaching agreement in the presence of faults, J. Assoc. Comput. Mach. 27 (2) (April. 1980) 228–234.
[4] I-SCOOP, Blockchain and the internet of things: the IoT blockchain opportunity and challenge [Online]. Available: https://www.iscoop.eu/blockchain-distributed-ledger-technology/blockchain-iot/, 2018.
[5] S. Nakamoto, Bitcoin: a peer-to-peer electronic cash system, White paper [Online]. Available: https://bitcoin.org/bitcoin.pdf, 2009.
[6] Nxt community, Nxt: a Peer-To-Peer Digital Socioeconomic System, White Paper, July. 2014.
[7] S. Popov, The tangle, White paper [Online]. Available: https://www.iota.org/research/academic-papers, 2018.
[8] BTC.com [Online]. Available: https://btc.com/.
[9] The evolution of ethereum [Online]. Available: https://blog.ethereum.org/2015/09/28/the-evolution-of-ethereum/.
[10] A. Churyumov, Byteball: a decentralized system for storage and transfer of value [Online]. Available: https://byteball.org/Byteball.pdf, 2016.
[11] SHA-2, In: English wikipedia [Online]. Available: https://en.wikipedia.org/wiki/SHA-2.

[12] L. Baird, The Swirlds Hashgraph Consensus Algorithm: Fair, Fast, Byzantine Fault Tolerance, White Paper, 2016 [Online]. Available: http://www.swirlds.com/developer-resources/whitepapers/.
[13] L.D. Xu, W. He, S. Li, Internet of things in industries: a survey, IEEE Trans. Ind. Inf. 10 (4) (Nov. 2014) 2233–2243.
[14] Li Yun, Xia Shichao, Yang Qianying, Wang Guoyin, Zhang Weiyi, Life Priority Driven Resource Allocation for WNV-Based Internet of Things, IEEE Internet of Things Journal (2020), https://doi.org/10.1109/JIOT.2020.3029175.
[15] G. BitFury, Proof of Stake versus Proof of Work, White paper, Sep. 2015.
[16] S.D. Lerner, DagCoin Draft, 2015 [Online]. Available: https://bitslog.files.wordpress.com/2015/09/dagcoin-v41.pdf.
[17] B. Cao, Y. Li, L. Zhang, L. Zang, S. Mumtaz, Z. Zhou, M. Peng, When internet of things meets blockchain: challenges in distributed consensus, IEEE Network (July. 2019) 1–7. Early Access.
[18] Igor Barinov, Viktor Baranov, Pavel Khahulin, POA network, White paper, Sep 28 [Online]. Available: https://github.com/poanetwork/wiki/wiki/POA-Network-Whitepaper, 2018.
[19] P4Titan, Slimcoin A Peer-To-Peer Crypto-Currency with Proof-Of-Burn "Mining without Powerful Hardware", White Paper, May 17, 2014 [Online]. Available: http://www.slimcoin.org.
[20] M. Castro, B. Liskov, Practical byzantine fault tolerance, in: Proceedings of the Third Symposium on Operating Systems Design and Implementation, New Orleans, USA, February 1999.
[21] S. Popov, O. Saa, P. Finardi, Equilibria in the tangle [Online]. Available: https://arxiv.org/pdf/1712.05385.pdf, 2017.
[22] S. Bag, S. Ruj, K. Sakurai, Bitcoin block withholding attack: analysis and mitigation, IEEE Trans. Inf. Forensics Secur. 12 (8) (Aug. 2017) 1967–1978.
[23] N. Papadis, S. Borst, A. Walid, M. Grissa, L. Tassiulas, Stochastic models and wide-area network measurements for blockchain design and analysis, in: Proc. IEEE Int. Conf. Commun., INFOCOM, Honolulu, USA, Apr. 2018.
[24] M. Li, J. Wang, A. Yang, W. Lu, Y. Zhang, L. Hou, J. Liu, Y. Xiang, R. Deng, CrowdBC: a Blockchain-based decentralized framework for crowdsourcing, IEEE Trans. Parallel Distrib. Syst. 30 (6) (June. 2019) 1251–1266.