The Impact of Block Parameters on the Throughput and Security of Blockchains

Elham Akbari
Cleveland State University
2121 Euclid Ave., USA
Cleveland, OH 44115
akbari.elh@gmail.com

Wenbing Zhao
Cleveland State University
2121 Euclid Ave. USA
Cleveland, OH 44115
wenbing@ieee.org

Shunkun Yang
Beihang University
School of Reliability and Systems Engineering
37 Xueyuan Rd, Beijing, China, 100191
ysk@buaa.edu.cn

Xiong Luo
University of Science and Technology Beijing
Beijing Key Lab of
Knowledge Engr. for Materials Sci.
30 Xueyuan Rd, Beijing, China, 100083
xluo@ustb.edu.cn

ABSTRACT

It has been well recognized that traditional blockchains have limited throughput. It is intuitive to achieve higher throughput by increasing the block size and shortening the block interval. In this paper, we study the security implications on doing so, and define the boundary for acceptable block sizes and block intervals. We define the security of the blockchain in terms of the stale block rate in the network and carry out an empirical study using a blockchain simulator to find the optimal block parameters (*i.e.*, size and interval). We show that it is possible to achieve sufficiently high throughput for a blockchain platform to be used for activities beyond cryptocurrency, such as state-level electronic voting.

CCS Concepts

• **Computing methodologies → Distributed computing methodologies → Distributed algorithms**

Keywords

Blockchain; Distributed consensus; Proof of work; Proof of stake; Proof of stake.

1. INTRODUCTION

The blockchain technology offers a revolutionary way of reaching consensus in an untrusted large-scale network [1]. The key idea is to convert the consensus problem into a randomized competition for solving a puzzle where the puzzle is hard enough that it takes tremendous resources to solve it and *typically* there is only a single winner for each round of competition [2]. Furthermore, the verification of the puzzle solution takes very little effort, which prevents a miner from cheating. This scheme is referred to as Proof of Work (PoW). Although this PoW-based consensus is much more robust, scalable, and requires very little assumption on the structure of the participants, which is a big contrast to traditional consensus algorithms [3-11], it does have one limitation, that is, the puzzle must not be made too simple so that it can be solved very quickly. Doing so would risk the occurrence of many winners in the same round, which defeats the purpose of using PoW to achieve consensus [12]. A direct consequence of this design is that the interval for different rounds of competition must not be too small, which in turn limits the maximum throughput of the blockchain system [13]. In Bitcoin, the block interval is set to be 10 minutes, which leads to a maximum throughput of 7 transactions per second as commonly claimed. In fact, we can easily estimate the maximum throughput based on the current block size limitation (1MB) and the average transaction size (500B) (i.e., 2000 transactions per block). The throughput is only less than half of what has been claimed, i.e., 3.33 transactions per second. As a comparison, the visa credit card network can process more than 470,000 transactions per second [13].

What block size should be adopted in Bitcoin has been hotly contested in the short history of Bitoin as reported in some online posts, such as the one at https://blocksdecoded.com/what-bitcoin-block-size/. Before the 1MB block size limitation was imposed in 2010, the maximum block can be as large as 36MB. In 2015, Gavin Andresen proposed to increase the block size as the network grows over time, which could be set to 8MB in 2016 and double the block size every 730 days until a ceiling is reached in 2036.

Ethereum, the second most popular cryptocurrency behind Bitcoin, uses a much smaller block interval. As shown at https://etherscan.io/chart/blocktime, the block interval is around 15 seconds, and the block size is much smaller, with most blocks being smaller than 20KB (https://etherscan.io/chart/blocksize).

While it is obviously that the block parameters (size and interval) have direct impact on the time it takes to solve the PoW puzzle and the transaction fee that can be collected, their security implications have not been well studied. In this paper, we perform a quantitative analysis on what range of block sizes and block intervals can be used with respect to the security of the

ICBCT'20, March 12–14, 2020, Hilo, HI, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7767-6/20/03…$15.00
https://doi.org/10.1145/3390566.3391673

blockchain platform using a blockchain simulator. We show that these two parameters would have direct impact on the stale block rate in the blockchain network. We show that it is possible to achieve sufficiently high throughput for a blockchain platform to be used for activities beyond cryptocurrency, such as state-level electronic voting in the United States [14].

2. BACKGROUND

In this section, we introduce some important parameters and performance characteristics of the blockchain and common attacks on the blockchain systems.

2.1 Blockchain Parameters

Stale block rate is the rate at which stale blocks are generated. “Stale blocks” refer to those blocks that are not included in the longest chain due to, for instance, contradiction or concurrency. Stale blocks are unfavorable to the security and performance of the blockchain as they initiate unwanted chain forks in the system. Chain forks negatively impact the growth of the main chain of the ledger and can cause bandwidth complications in the network. But above all, the presence of a large number of stale blocks increases the ability of dishonest nodes in performing fraudulent activities such as selfish mining, as explained later.

Block interval is one of the most important parameters of the blockchain systems and is determined by the delay at which data is appended onto the ledger. A smaller block interval would naturally lead to higher throughput, but at the expense of a higher likelihood of generating “stale blocks.” Adjusting the block interval implies changing the difficulty level of the PoW puzzle. The difficulty of the PoW puzzle is conversely correlated to the rate at which stale blocks can be generated. This in turn infers that adjusting the difficulty of the puzzle can directly impacts the capability of the dishonest nodes in attacking the network through tampering with the longest chain of the ledger.

Block size determines the number of transactions that can be collected within each block. Accordingly, the maximum block size regulates the throughput of the blockchain system. The larger the size of the block is, the slower the propagation speed and the higher the slate block rate will be. Therefore, if one is to improve the throughput of the system, reducing the security of the system will be inevitable.

Information propagation mechanism is the mechanism by which the blocks are broadcast to various network nodes. The broadcast scheme that is determined by the block request management system directly influences the scalability and robustness of the ledger. The most widely used propagation scheme is an advertisement-based management system. In such a method, as soon as a node receives data from another node, it will advertise the hash of that block to other connections in the network. In case one of the nodes has not already received that particular data, it will request for the content of the block.

Mining power is the ratio of the power of the dishonest portion of the network to that of the entire network.

2.2 Common Attacks on Blockchain

There are three typical attacks on a blockchain system: (1) double spending, (2) selfish mining, and (3) eclipse attack.

For cryptocurrencies, *double spending* is referred to as the case where a certain number of coins are spent in more than one transaction. There are three conceivable ways that double spending could happen: (1) When two contradictory transactions are submitted to the network in quick succession, a race attack is occurred. Obviously, only one of them that involves in the longest chain will eventually go through. (2) When one transaction is pre mined into a new block, but it is not released until the very same coins are used for another new transaction. This method, which is called Finney attack, may result in the invalidation of the first transaction, if successful. (3) When more than 51% of the overall computing power in the network is devoted to undoing a transaction and instead putting through a preferred transaction, the attack is also referred to as 51% attack. It is important to note that in each of the above attacks, the person who originally submits a transaction is the beneficiary and therefore the actual fraudster.

Selfish mining occurs when a team of dishonest miners collude to augment their mining reward revenue [15]. In such a scenario miners can potentially earn more reward by concealing the newly produced blocks from the main chain and creating a distinct fork. In Bitcoin, the incentives the miners receive are proportional to their mining output. Hence, even if big groups of miners attempt colluding, they cannot receive more coins combined than what they would individually and collectively generate in the public ledger. Nonetheless, if dishonest nodes conceal the new blocks and make them available only in their private network, they can raise their share of the network’s overall reward. Selfish mining may jeopardize the decentralization nature of blockchains.

Selfish mining attacks can have profound effects on the integrity of blockchain system. When successful, dishonest adversaries can easily turn into more profitable nodes than the honest nodes. Profits from selfish mining can arise if more computational power is utilized by the adversaries. This can make the attacks exponentially more effective, until to a point where over 50% of the power in the network is held in favor of the attackers. This can ultimately force regular nodes out of the network. In such a case, the dishonest portion of the network would be not only able to gather all the block rewards, but also to block any transactions from being processed fairly.

Eclipse attack is another deceitful activity in blockchains in which a dishonest node takes control of the victim’s inward and outward connections, hence separating the victim from the rest of the nodes in the network [16]. The invader can then block the victim’s visibility of the network and obligate them to spend their computing power on viewing an outdated version of the blockchain network, or even worse divert the power to the advantage of his/her own iniquitous activities. Other than interrupting and damaging the integrity of the blockchain network, eclipse attacks could be the onset of and escalate other potential attacks such as selfish mining.

3. RELATED WORK

Increasing the block size is one way to increase the throughput of the blockchain system. For example, Bitcoin Cash, which is a fork from Bitcoin, imposes an 8MB limit (although the actual block sizes are typically much smaller than this limit). On the other hand, shortening the block interval could also increase the system throughput, as exemplified by Ethereum, which has a block interval of about 15 seconds. Some Bitcoin forks, such as Litecoin, uses a block interval of 2.5 minutes, and Dogecoin, uses a block interval of 1 minute.

In [13], the authors reported a theoretical study on the tradeoffs between the blockchain system security and the block generation rate, which we refer to as block interval. They introduced a new security property called chain growth. This property defines the minimum growth rate for the chain as viewed by honest miners. They hypothesize that it is the best interest for an adversary to reduce the system throughput (or to enlarge the block interval). They performed an analysis with respect to the proposed chain growth, and two previously proposed measures, common prefix and chain quality, as a function of block intervals. The common prefix is used to assess whether or not two miners have the same view of the blockchain. The chain quality describes the extend of the chain as accepted by honest miners that contains sequences of adversarial blocks. The most interesting result is that, unlike the prediction made by an earlier model [17], which would lead to a total insecurity of the system (that is, the system will break down even in the presence of a very small fraction of adversaries) if the block interval is on par with the block propagation time, the system can actually tolerate the presence of up to 1/3 Byzantine faulty miners under the same condition. It is interesting to note that the block interval chosen by Ethereum is quite close to the average block propagation time (12.6 seconds) as reported in [18].

Different from the above paper, we engage an empirical simulation study with a simulator that mimics the Bitcoin operation environment using higher level, easier-to-understand security measures. Instead of only considering the block interval, we also use the block size as an important parameter. In [13], the authors assumed that the block propagation time is linearly proportion to the block size. We do not think that this is necessarily true because the Internet backbone has very high bandwidth and the world-wide block propagation time might be dominated by the queuing time at the router (which is not proportional to the block size, but rather impacted by the congestion degree of the Internet) instead of transmission time (which it is proportional to the block size).

Other than adjusting the block size and the block interval, several orthogonal approaches have been proposed. The essence of all these approaches is to rely on the transactions placed on the main blockchain to ensure the security of transactions elsewhere.

In [19], the authors proposed to use local blockchains to increase the scalability of blockchain. In their scheme, the root of each local chain is placed on the main blockchain so that the linkage between the main blockchain and the local chain is established. However, the local chains are not automatically protected by the main blockchain platform.

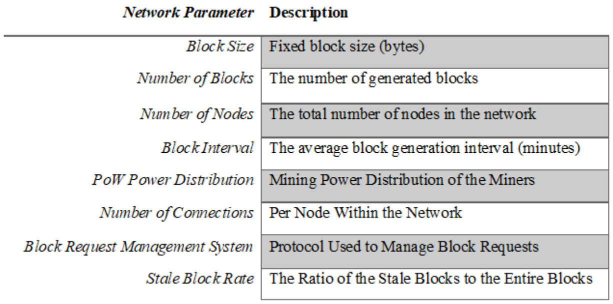
Previously, we proposed a hierarchical blockchain architecture to facilitate electronic voting [14]. The hierarchy would align with the voting scale with the lowest level being at the precinct. Higher levels could be county, state, and the national level. One blockchain would be used for each precinct, and then these blockchains would be linked according to the hierarchy.

Neither [19] nor our previous proposals [14] ensures the same degree of security for off-chain transactions as those in the blockchain. In another paper [20], we proposed to establish a strong linkage between the off-chain transactions and those placed on the blockchain, thereby, those off-chain transactions can have the same degree of security as those on-chain.

Another approach is to establish a payment channel among two or more parties where transactions sent in the channel will not be placed on the main blockchain, hence, they are much faster [21]. The channel itself is protected with transactions placed on the blockchain. These transactions ensure the integrity of the transactions in the channel. The focus of this scheme is to protect the integrity of the agreement between different payment channel parties instead of recording the off-chain transactions permanently.

4. QUANTITATIVE STUDY ON THROUGHPUT AND SECURITY

In this section, we leverage a blockchain simulator developed by researchers at ETH [22] to study the scalability and the throughput of the Bitcoin ledger. Due to the trade-off between the throughput and security of blockchain systems, it is also imperative to study the security concerns of blockchain systems. Stale blocks are known to pose serious danger to the integrity and security of distributed ledgers. Accordingly, we use block interval and block size as the inputs of the blockchain simulator to determine the stale block rate in the system. Interestingly, block interval and block size are also key parameters when it comes to calculation of the system throughput.



Network Parameter	Description
Block Size	Fixed block size (bytes)
Number of Blocks	The number of generated blocks
Number of Nodes	The total number of nodes in the network
Block Interval	The average block generation interval (minutes)
PoW Power Distribution	Mining Power Distribution of the Miners
Number of Connections	Per Node Within the Network
Block Request Management System	Protocol Used to Manage Block Requests
Stale Block Rate	The Ratio of the Stale Blocks to the Entire Blocks

Figure 1. The important network parameters that can be captured in the blockchain simulator used in this work

4.1 Blockchain Simulator

Since a real world implementation with thousands of nodes is extremely challenging in many cases, a powerful simulator can be of vital importance for realistically studying the blockchain performance as a function of network parameters. Recent studies on blockchain systems suggest that there is a trade off between the performance and security of PoW based blockchains. Therefore, it is extremely helpful to have a unified framework that can capture such trade offs as a function of different network parameters. The novel quantitative framework introduced in [22] can analyze the security and performance implications of various parameters of PoW blockchains. Taking advantage of such framework, not only the security properties of well known PoW platforms such as Bitcoin, Ethereum, and Litecoin can be examined, but also important blockchain parameters can be adjusted to branch out into other similar platforms. This framework is comprised of two main elements, a blockchain instance and a blockchain security model. A blockchain instance is a PoW blockchain that is represented by a certain set of network parameters, such as block generation time, block size, network delays, etc. To convincingly analyze any blockchain instance, a simulator can be used to replicate the blockchain network and consensus layers through the implementation of advertisement based data transmission. One of the key outputs of a blockchain platform is the rate at which the stale blocks are generated.

The list of some of the important blockchain parameters that can be captured by the ETH simulator are summarized in Fig.1.

In the simulator, assigning a new block to a miner is determined based upon the block interval. In compliance with the existing

PoW blockchains, the simulator assumes that typical miners start mining as soon as they receive a block. Also it is assumed that potential forks get automatically resolved based on the longest chain rule. After resolving any fork, the blocks that are not attributed to the main chain of the network are constituted as stale blocks. Since the difficulty variations between various blocks are not considered in the simulator, the lengths of the chains are just defined and calculated based on the number of blocks forming each chain.

In the communication protocol between the nodes, the channels are formed directly in between every pair of nodes. This way any intermediary machine such as routes are sidestepped completely. Each channel, in this context, would have two main features, bandwidth and latency. In order to credibly take the effect of network latencies in the simulator, the developers have employed the global IP latency statistics from Verizon. Furthermore, in order to accurately capture the bandwidth of the network, testmy.net has been utilized to obtain the bandwidth distributions. However, since the main intent of the simulator is to investigate the effect of the network parameters such as the block size and the block interval, it does not capture transaction propagation which has no correlation with the above mentioned parameters.

The simulator differentiates miners from the typical nodes of the network. The geographical node distribution of the network is extracted from blockchain.info. Based on this distribution, around 52% of the nodes are located in Europe while North America contributes about 39% of the nodes. The remaining nodes in percentile order are traced to be in Asia Pacific, Australia, Japan, and South America. On the other hand, the distribution of the miners is quite different. Asia pacific possesses a share of about 71%, while North America and Europe only contribute about 24% and 5%, respectively [22]. It should be noted that such distribution which is correlated to the Bitcoin blockchain is merely utilized to replicate a real world implementation.

4.2 Simulation Conditions

The simulations are conducted based on the assumption that the dishonest nodes cannot potentially utilize more than 30% of the overall mining power [23]. We perform the simulations for block sizes ranging from 1 KB up to 25 MB, given different block intervals ranged between 1 seconds and 30 minutes. The number of generated blocks and the total number of nodes in the network are both considered to be 100 in all the simulations. Based on such combination, each simulation run takes about 70 seconds to complete which is orders of magnitude faster than an actual implementation if we were to really execute the scenario in real world. As a reference, it should be mentioned that simulation runs with the number of blocks/nodes of 500/100, and 100/500 would take about 300 seconds and 900 seconds to finish, respectively. A 500/500 combination for number of blocks/nodes also requires longer than an hour to finish for every run. On the other h and, the simulations suggest that they all result in comparable outcome, stale block rate, to that of 100/100 combination (see Fig.2). Since we are to perform over 50 runs, the latter is used to save considerable amount of time.

4.3 Simulation Results

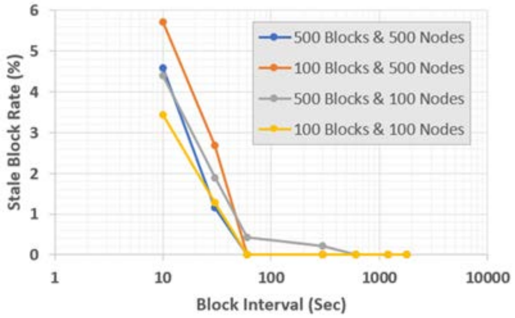


Figure 2. Stale block rate as a function of block interval for a block size of 10 KB and for different combinations for the number of blocks and nodes

Simulations were performed for six different block sizes 1 KB, 10 KB, 100 KB, 1 MB, 10 MB, and 25 MB, while measuring the stale block rate for each of the following block intervals 1, 10, 30, 60, 300, 600, 1200, and 1800 seconds. This accounts for 48 individual scenarios. By conducting these simulations we aim at studying the impact of block interval and block size on the stale block rate. Stale block rate is very pivotal to determining the security of the blockchain network. A smaller stale block rate is more desired and typically represents a more secure system.

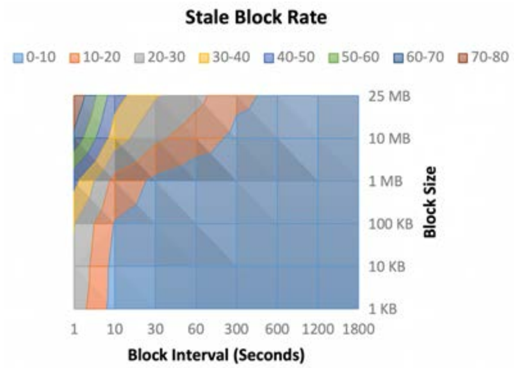
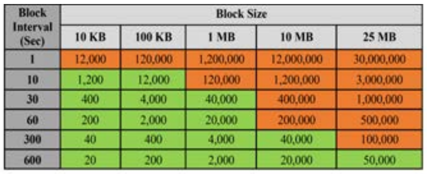


Figure 3. The color map of stale block rate as a function of block size and block interval

As a general trend, the simulation results as shown in Fig. 3 suggest that the stale block rate is inversely correlated with the block interval. Moreover, at a constant block interval larger block size results in a higher value of stale block rate. Accordingly, it is safe to conclude that increasing the block interval and block size will improve and degrade the security, respectively.

If we use the typical transaction size of 500 bytes, and if we further assume that we can tolerate up to 10% stale block rate, we can estimate the maximum throughput that can be achieved in allowable block size and interval combinations. The results are summarized in Fig. 4, where the green color cells indicate allowable combinations and the orange cells would cause a stale block rate of over 10%. We omitted the result for 1KB block size because it is not practical. As can be seen, for the 25MB block size, the smallest block interval is 600 seconds (i.e.,10 minutes, which is identical to what is used by Bitcoin). The combination of 25MB block size and 600 seconds block interval gives the maximum possible throughput of 50,000 transactions per 600 seconds (i.e., 83 transactions per second). All other combinations with smaller block sizes would give lower maximum throughput.



Block Interval (Sec)	Block Size				
	10 KB	100 KB	1 MB	10 MB	25 MB
1	12,000	120,000	1,200,000	12,000,000	30,000,000
10	1,200	12,000	120,000	1,200,000	3,000,000
30	400	4,000	40,000	400,000	1,000,000
60	200	2,000	20,000	200,000	500,000
300	40	400	4,000	40,000	100,000
600	20	200	2,000	20,000	50,000

Figure 4. Throughout of the blockchain voting system for different combinations of block interval and block size. The number of transactions shown in each cell is per 10 minutes (600 seconds)

5. CONCLUSION

In this paper, we presented an empirical study on the relationship between the block parameters and the stale block rate, which would directly impact the security of the blockchain system. The simulator we used mimics the Bitcoin operating environment. Our study provides boundary block parameter combinations so that the stale block rates are kept below 10%. Among the parameters that we have experimented, a block size of 25MB with a block interval of 600 seconds appears to give the highest throughput at 83 transactions per second. A block size of smaller than 1MB appears to be not practical, even though the smaller block size could allow the use of smaller block intervals. For future work, we plan to frame this problem as a reinforcement learning problem [25,26] to discover the best parameters that strike a balance between performance and security in blockchain systems.

6. ACKNOWLEDGMENTS

This paper is partially based on the master's thesis [24] of the first author who has the copyright. This work was supported in part by the United States Department of Energy Award DE-FE0031745, the National Natural Science Foundation of China [grant 61672080], JWYY [grant 41402020502], and NASFC [grant 2016ZD51031].

7. REFERENCES

[1] Nakamoto, S. Bitcoin: A peer-to-peer electronic cash system. http://www.bitcoin.org.

[2] Zhao, W., Yang, S., & Luo, X. (2019, March). On Consensus in Public Blockchains. In *Proceedings of the 2019 International Conference on Blockchain Technology* (Honolulu, Hawaii, USA, March 2019). ACM, 1-5.

[3] Zhao, W. 2014. *Building dependable distributed systems*. John Wiley & Sons.

[4] Zhao, W., Melliar-Smith, P. M., & Moser, L. E. 2012. Low latency fault tolerance system. *The Computer Journal*, 56, 6 (June 2012), 716-740.

[5] Castro, M., & Liskov, B. 2002. Practical Byzantine fault tolerance and proactive recovery. *ACM Transactions on Computer Systems* 20, 4 (Nov. 2002), 398-461.

[6] Zhao, W. 2016. Optimistic byzantine fault tolerance. *International Journal of Parallel, Emergent and Distributed Systems* 31, 3 (2016), 254-267.

[7] Zhao, W. 2014. Application-aware byzantine fault tolerance. In *Proceedings of the IEEE 12th International Conference on Dependable, Autonomic and Secure Computing* (Dalian, China, August 24-27, 2014). IEEE, 45-50.

[8] Babi, M., & Zhao, W. 2017. Towards Trustworthy Collaborative Editing. *Computers* 6, 2 (March 2017), 13.

[9] Zhao, W. 2016. Performance optimization for state machine replication based on application semantics: a review. *Journal of Systems and Software*, 112 (February 2012), 96-109.

[10] Zhang, H., Chai, H., Zhao, W., Melliar-Smith, P. M., & Moser, L. E. 2012. Trustworthy coordination of Web services atomic transactions. *IEEE Transactions on Parallel and Distributed Systems*, 23, 8, (2012), 1551-1565.

[11] Chai, H., Zhang, H., Zhao, W., Melliar-Smith, P. M., & Moser, L. E. 2013. Toward trustworthy coordination of Web services business activities. *IEEE Transactions on Services Computing*, 6, 2, (2013), 276-288.

[12] Wang, W., Hoang, D. T., Xiong, Z., Niyato, D., Wang, P., Hu, P., & Wen, Y. (2019). A survey on consensus mechanisms and mining management in blockchain networks. *IEEE Access*, 7, (2019), 22328-22370.

[13] Kiayias, A., & Panagiotakos, G. 2015. Speed-Security Tradeoffs in Blockchain Protocols. *IACR Cryptology ePrint Archive*, (2015), 1019.

[14] Akbari, E., Wu, Q., Zhao, W., Arabnia, H. R., & Yang, M. Q. 2017. From Blockchain to Internet-Based Voting. In *Proceedings of the International Conference on Computational Science and Computational Intelligence* (Las Vegas, NV, USA, December 14-16, 2017). IEEE, 218-221.

[15] Sapirshtein, A., Sompolinsky, Y., & Zohar, A. 2016. Optimal selfish mining strategies in bitcoin. In *Proceedings of the International Conference on Financial Cryptography and Data Security* (February 2016). Springer, Berlin, Heidelberg. 515-532.

[16] Nayak, K., Kumar, S., Miller, A., & Shi, E. 2016. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In *Proceedings of the IEEE European Symposium on Security and Privacy* (2016, March). IEEE. 305-320.

[17] Garay, J., Kiayias, A., & Leonardos, N. (2015). The bitcoin backbone protocol: Analysis and applications. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (April 2015). Springer, Berlin, Heidelberg. 281-310.

[18] Decker, C., & Wattenhofer, R. 2013. Information propagation in the bitcoin network. In *Proceedings of the IEEE P2P* (September 2013). IEEE. 1-10.

[19] Poon, J. and Dryja, T. 2016. The bitcoin lightning network: Scalable offchain instant payments, Lightning Labs, Tech. Rep., Nov. 2016

[20] Zhao, W., Yang, S., & Luo, X. 2020, Secure Hierarchical Processing and Logging of Sensing Data and IoT Events with Blockchain. In *Proceedings of the International Conference on Blockchain Technologies* (Hilo, Hawaii, HL, March 2020). ACM.

[21] Antonopoulos, A. M. 2014. *Mastering Bitcoin: unlocking digital cryptocurrencies*. O'Reilly Media, Inc.

[22] Gervais, A., Karame, G. O., Wüst, K., Glykantzis, V., Ritzdorf, H., & Capkun, S. 2016. On the security and performance of proof of work blockchains. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security* (October 2016). ACM. 3-16.

[23] Eyal, I., & Sirer, E. G. (2018). Majority is not enough: Bitcoin mining is vulnerable. *Communications of the ACM*, *61*(7), 95-102.

[24] Akbari, E. 2018. From Blockchain to Internet-Based Voting (Masters dissertation, Cleveland State University).

[25] Chen, M., Li, Y., Luo, X., Wang, W., Wang, L., & Zhao, W. (2019). A novel human activity recognition scheme for smart health using multilayer extreme learning machine. *IEEE Internet of Things Journal*, 6(2), 1410-1418.

[26] Luo, X., Sun, J., Wang, L., Wang, W., Zhao, W., Wu, J., Wang, J. H., & Zhang, Z. (2018). Short-term wind speed forecasting via stacked extreme learning machine with generalized correntropy. *IEEE Transactions on Industrial Informatics*, 14(11), 4963-4971.