Journal Pre-proof

Unintentional Forking Analysis in Wireless Blockchain Networks

Qilie Liu, Yinyi Xu, Bin Cao, Lei Zhang, Mugen Peng

PII: S2352-8648(20)30292-3

DOI: https://doi.org/10.1016/j.dcan.2020.12.005

Reference: DCAN 265

To appear in: *Digital Communications and Networks*

Received Date: 3 March 2020

Revised Date: 30 November 2020

Accepted Date: 9 December 2020

Please cite this article as: Q. Liu, Y. Xu, B. Cao, L. Zhang, M. Peng, Unintentional Forking Analysis in Wireless Blockchain Networks, *Digital Communications and Networks*, https://doi.org/10.1016/j.dcan.2020.12.005.

Digital Communications and Networks(DCN)

Journal Pre-proof

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/dcan

Unintentional Forking Analysis in Wireless Blockchain Networks

Qilie Liu[1], Yinyi Xu[1], Bin Cao*[2], Lei Zhang[3], and Mugen Peng[2]

[1]**College of Communications and Information Engineering and Chongqing Key Lab of Mobile Communications Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China**
[2]**State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China.**
[3]**James Watt School of Engineering, University of Glasgow, Glasgow, G12 8QQ, U.K.**

Abstract

Forking problem plays a key role in the security issue, which is a major concern in the blockchain system. Although many works studied on the attack strategy, consensus mechanism, privacy-protecting and security performance analysis, most of them only address the intentional forking caused by a malicious attacker. In fact, without any attacker, unintentional forking still remains due to transmission delay and failure, especially in wireless network scenarios. To this end, this paper investigates the reason to generate unintentional forking, and derives the forking probability expression in Wireless Blockchain Networks (WBN). Furthermore, in order to illustrate the unintentional forking on blockchain system, performance in terms of resource utilization rate, block generation time, and Transaction Per Second (TPS) are investigated. The numerical results show that the target difficulty of hash algorithm in generating a new block, the delay time of broadcasting, the network scale, and the transmission failure probability would affect the unintentional forking probability significantly, which can provide a reliable basis for avoiding forking to save resource consumption and improving system performance.

KEYWORDS:
Wireless blockchain network, Unintentional forking, Proof-of-Work, Stochastic theory.

1. Introduction

Blockchain is a Peer-to-Peer (P2P) distributed ledger technology, which has been identified as one of the most devastating technologies of this century, and it has attracted widespread attention in industry and academia [1]. Blockchain has the advantages of decentralization, traceability, transparency, and high security. These advantages make blockchain become the underlying technology for cryptocurrency systems such as Bitcoin and Ethereum with billions of dollars in market capitalization. In recent years, the concept of building trust and consensus in a distributed environment has made the wide application of blockchain in scenarios far beyond the scope of cryptocurrencies, such as 5G heterogeneous networks [2, 3] and the IoT ecosystem [4] (such as edge computing [5, 6], e-Health system [7] and supply chain [8]).

The blockchain architecture consists of three parts, namely, transaction, block and chain. In the blockchain, any valuable information can be broadcast to the P2P networks as a transaction. The unit where transactions are stored is called a block, and each block is uniquely identified by its hash value. A hash of the previous block is written to the next block, forming a block chain. In order to realize decentralization as well as safety, consensus mechanisms, such as Proof-of-Work (PoW), Proof-of-Stake (PoS), RAFT and Byzantine Fault

*The corresponding author: Bin Cao (email: caobin@bupt.edu.cn). 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 Teachers Project, in part by the Chongqing Science and Technology Innovation Leading Talent Support Program (CSTCCXLJR-C201710), and in part by Chongqing Technological Innovation and Application Development Projects (cstc2019jscx-msxm1322).

Tolerance (BFT), are designed for blockchain in a distributed manner [9-12]. The consensus mechanism enables network nodes to add new blocks to the chain efficiently and safely, and all nodes have the same transaction information. Since PoW is the most famous and widely used consensus mechanism, we use PoW as an example in the rest of this paper, and this work can be extended easily using other PoX mechanisms.

In blockchain systems, all nodes start a new mining with the previous block connection as a chain, and the chain with the longest and most blocks is called the main chain [13]. For various reasons discussed in [14], when different nodes mine on different blocks to generate the new block, a chain would fork as two or many, which is called as forking [15]. Forking is one of the most significant problems in blockchain, which would decline the performance and cause security issue, and thus it has been widely addressed in previous works [16-18]. However, most of the related research has focused on the forking problem incurred by malicious attackers, which can be categorized as intentional forking. In contrast, the other is unintentional forking that occurs naturally during the operation of the blockchain system without any attackers. This problem has not been solved well, and it still produces a deteriorated effect on performance and security like the intentional one.

PoW is a computational power competition. In this competition process, the first node (or miner) to solve hash puzzle is the winner who can generate the new block and broadcast it to the whole network. Since the main chain has the most number of cumulative blocks, it has the lowest probability of being abandoned. As a result, any rational node would prefer on mining work on the main chain. However, in the blockchain system, especially in wireless network scenarios, some nodes may not know mining result immediately due to transmission delay or failure. Thus, these nodes would keep on mining on the previous block instead of the new one and occur the unintentional forking. Like intentional forking, even if there is no malicious attacker, the unintentional forking have a certain probability to occur, which wastes the work of the blockchain system and reduces the overall computing power of the system. Consequentially, this unintentional forking will make the overall growth rate and the system performance deteriorated, such as the lower confirmation delay. In addition, the higher frequency of unintentional forking and slower overall growth rate will also indirectly increase the probability successfully attacks launched by adversarial attackers, thus affecting the security level of system and making it more vulnerable.

Inspired by the above observations, this paper aims to investigate the unintentional forking in Wireless Blockchain Networks (WBN). First, considering the impact of transmission delays and failures, we discussed the causes of unintentional forking. Second, we formulate the unintentional forking as a stochastic problem and derive the close form of unintentional forking probability. Then, we analyze the system performance to investigate the negative effect of unintentional forking. Finally, the numerical results validate the rightness and effectiveness of our analytical model.

2. Related work

In recent years, the performance and security of the blockchain have attracted widespread attention from the academic community. Y. Sun et al. [22] discussed the relationship between communication

throughput and transaction throughput in a blockchain-based wireless IoT system and proposed an optimal communication node deployment algorithm. In [23], the authors analyzed the impact of unstable network load on the performance and security in DAG-based blockchains by using Markov processes. Gervais et al. [24] analyzed the impact of block size, block generation interval, and various network parameters on the security of PoW-based blockchains. Sapirshtein A et al. [25] expanded the selfish mining attack model in Bitcoin, researched the optimal selfish mining strategy for attackers, and found that even if the computational power possessed by the attacker is less than 25% of the whole network, it can still launch selfish attacks to obtain benefits. Heilman et al. [26] studied the eclipse attack in Bitcoin in detail and proposed countermeasures to increase the difficulty of the eclipse attack and ensure the security of the blockchain network.

Ref. [27] discussed the mechanism and characteristics of forking in distributed Bitcoin's P2P network, and pointed out that there was no obvious correlation between the duration of the forking conversion, the number and size of network partitions caused by the forking. B. Liu et al. [14] discussed the reasons for forking in the blockchain. They believed that the blockchain forking was caused by competition between miners in the network and uncertain block transmission delays. A PvScheme scheme for P2P receiving block probability verification is proposed to reduce the occurrence of blockchain forking by reducing the block transmission delay. C. Decker et al. [28] discussed the relationship between block transmission delay and forking in the blockchain network from the perspective of the network, and proposed that the forking phenomenon in the blockchain can be effectively reduced by changing the Bitcoin protocol. Y. Shahsavari et al. [29] introduced the reasons for the forking in the Bitcoin network, and discussed the relationship between block size, block transmission delay, network bandwidth, etc, and derived the probability of blockchain forking. They pointed out that the forking in the blockchain was independent of the average number of node connections.

However, most of the related work has focused on the intentional forking problem caused by malicious attackers. In fact, even without any attacker, the forking problem still occurs in the natural operation of the blockchain system. Although some related work has studied unintentional forking in the blockchain system, most of them only considered unintentional forking due to transmission delay, and there are few studies on the performance from the perspectives of transmission delay and failure. As far as we know, considering the effects of transmission delays and failures, mathematical analysis and discussion of the performance of actual wireless scenarios is the first task.

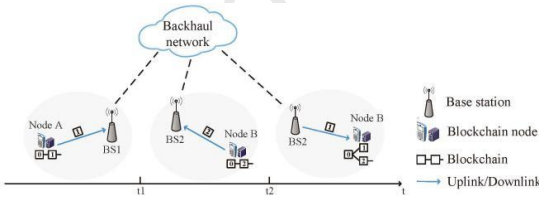
3. System Model and basic principles

3.1. System model

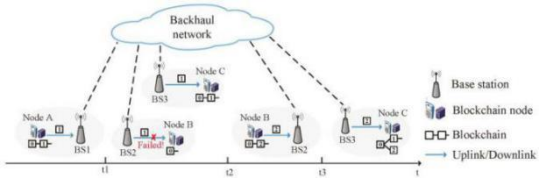
We consider WBN based on the PoW consensus mechanism and cellular networks, which consists of N mining nodes which are randomly distributed, and each node is associated with the closest base station through a wireless channel for connection. Let the set of mining nodes in WBN be $N = \{1, 2, \ldots, N\}$, and computational power of the ith node ($i \in N$) be r_i. In WBN, the process of generating a

valid block and updating the entire network ledger is illustrated as follows. 1) When a node finds a right hash value for the target, a new block is generated and connected to its local ledge. Meanwhile, the new block would be broadcast to the associated base station through uplink immediately. 2) The base station sends the new block to other associated nodes within the same cell and the remaining base stations in WBN through downlink and backhaul, respectively. 3) Then, the remaining base stations broadcast the new block and the whole WBN receive it. 4) Finally, each node updates its local ledger accordingly and continue new mining work after this block.

In this process, if a transmission delay or failure occurs, some nodes cannot receive the new broadcast block to be updated, and thus the mining work would on different block in a forking manner (i.e., some nodes work on the new block and the rest work on the previous block). Consequentially, unintentional forking is caused. Fig. 1 shows a typical scenario of unintentional forking, in which *block* 0 is the genesis block.



(a) forking caused by transmission delay



(b) forking caused by transmission failure

Fig. 1: A typical unintentional forking scenario in wireless blockchain networks

3.2. *Proof-of-Work*

PoW is a consensus mechanism based on computational power competition, which is proposed for Bitcoin network first. According to [20], we can know that in the PoW-based blockchain system, the time duration T_i for node i generating a new valid block is related to its own computational power r_i, which follows a negative exponential distribution as

$$P\{T_i \le t\} = 1 - \exp(-\frac{r_i t}{D}), \tag{1}$$

where D is the target difficulty value.

Consider a WBN with N nodes, then the system generates a valid block whose time period T is the minimum time among all nodes, which is shown as follows [20]:

$$P\{T \le t\} = 1 - \exp(-\frac{\sum_{i=1}^{N} r_i}{D} t). \tag{2}$$

3.3. *The coverage in cellular networks*

In order to investigate the impact of transmission delay and failure on unintentional forking, we need to model cellular networks to obtain transmission link coverage probabilities. Most of the previous work evaluates the coverage probability of cellular networks based on a grid structure, in which mobile users are randomly placed deterministically. However, in actual cellular networks, this method is highly desirable and intractable. In order to obtain accurate coverage probability and better evaluate the impact of transmission failures on unintentional forking in the blockchain system, we refer to the system model and analysis results in [21], which considers the interference of the whole network based on stochastic geometry. The base station is deployed randomly following homogeneous Poisson Point Process (PPP) with intensity γ in the Euclidean plane, in which each mining node communicates with

its nearest base station, and all other base stations are interference sources. The interference power follows the general statistical distribution g, the noise power is additive with constant value σ^2. Using the standard power loss propagation model, the path loss exponent α is generally larger than 2, and the transmitting power is constant with value $1/\mu$. For the random channel effects, the target base station and the target node only experience a Rayleigh fading with mean 1. The coverage probability is defined as the probability that a randomly chosen node can achieve the target **SINR** S.

Then, the coverage probability P_c of a typical randomly located mining node to a base station with a distance d is shown as follows [21]:

$$P_c \overset{def}{=} P_c(S,\gamma,\alpha) = \pi\gamma\int_0^{\infty} e^{-\pi\gamma\upsilon\beta(S,\alpha)-\mu S\sigma^2\upsilon^{\alpha/2}} d\upsilon, \quad (3)$$

$$\beta(S,\alpha) = \frac{2(\mu S)^{\frac{2}{\alpha}}}{\alpha} E[g^{\frac{2}{\alpha}}(\Gamma(-2/\alpha,\mu Sg) - \Gamma(-2/\alpha))], \quad (4)$$

where υ denotes the square of the distance d, $\Gamma(a,x) = \int_x^{\infty} t^{a-1}e^{-t}dt$ denotes the incomplete gamma function, and $\Gamma(x) = \int_0^{\infty} t^{x-1}e^{-t}dt$ denotes the standard gamma function, respectively.

4. Probability of unintentional forking

In WBN, when a node generates a new block, the node would broadcast it to the whole network. Due to transmission delay and failure, perhaps, another node generates another block before receiving it successfully. As a result, two different blocks are generated at the same blockchain height without any malicious attackers, which is unintentional forking.

As mentioned before, unintentional forking can be categorized into two cases. The first case occurs during the new block broadcasting period, another node generates the other block when the first new one has not been received, which is caused by transmission delay. The second case occurs due to transmission failure, the new block fails to transmit to some nodes, and thus they do not know it and keep on mining to generate another block. In summary, the unintentional forking probability is defined as follows:

$$P\{forking\} = P\{case\ 1\} + P\{case\ 2\}. \quad (5)$$

Next, we will discuss the two cases in details.

4.1. case 1: forking caused by transmission delay

The first node which generates a valid block in WBN is defined as k, $k \in N$, and the generated block is recorded as *block a*. During the transmission period of *block a*, if any other node (denoted as l, $l \in N$ and $l \neq k$) generates another block (recorded as *block b*), unintentional forking would occur.

Therefore, we can define the unintentional forking probability in *case 1* as follows:

$$P\{case\ 1\} = P\{T_k \le T_l < T_k + h\}, \quad (6)$$

where T_k is the time that node k generates *block a*, T_l is the time that node l generates *block b*, and h is the transmission delay for block broadcasting.

Based on conditional probability, equation (6) can be rewritten as

$$P\{case\ 1\} = P\{T_k \le T_l\}P\{T_l < T_k + h \mid T_k \le T_l\}. \quad (7)$$

According to the introduction of PoW in

Section III, T_l follows an exponential distribution, based on the memoryless property of exponential distribution, we have

$$\begin{aligned} P\{case\ 1\} &= P\{T_k \le T_l\} \cdot (1 - P\{T_l \ge T_k + h \mid T_l \ge T_k\}) \\ &= P\{T_k \le T_l\} \cdot (1 - P\{T_l \ge h\}) \\ &= P\{T_k \le T_l\} \cdot P(T_l < h\}. \end{aligned} \tag{8}$$

Furthermore, since the distribution of T_l is determined by the rest of nodes (or say the rest of computational power) except node k, T_k is also affected by the node which generates *block a*. As a result, $P\{case\ 1\}$ can be expressed as

$$\begin{aligned} P\{case\ 1\} &= \sum_{k=1}^{N} P\{T_k \le T_l\} P\{T_l < h\} \\ &= \sum_{k=1}^{N} P\{T_k \le T_{-k}\} P\{T_{-k} < h\}. \end{aligned} \tag{9}$$

where T_{-k} represents the time that all nodes of the WBN expect node k generates a valid block.

According to the following theorem:

Theorem *if* X_1 *and* X_2 *are independent exponential random variables with respective means* $1/\lambda_1$ *and* $1/\lambda_2$, *then* $P\{X_1 \le X_2\} = \dfrac{\lambda_1}{\lambda_1 + \lambda_2}$,

we can get that

$$\begin{aligned} P\{T_k \le T_{-k}\} &= P\{T_k \le \min(T_1, T_2, ..., T_{k-1}, T_{k+1}, ...T_N,)\} \\ &= \frac{r_k}{\sum_{i=1}^{N} r_i}. \end{aligned} \tag{10}$$

In summary, we can have the expression of unintentional forking probability in *case* 1 as

$$P\{case\ 1\} = \sum_{k=1}^{N} \frac{r_k}{\sum_{i=1}^{N} r_i} [1 - \exp(-\frac{\sum_{i=1}^{N} r_i - r_k}{D} h)], \tag{11}$$

where $h = \dfrac{L_1}{R_{up}} + \dfrac{L_1}{R_{backhaul}} + \dfrac{L_1}{R_{down}}$, and L_1 denotes the block payload, R_{up} denotes uplink rate, $R_{backhaul}$ denotes backhaul rate, and R_{down} denotes downlink rate.

4.2. case 2 : forking caused by transmission failure

If there is no unintentional forking occurred in *case* 1, the unintentional forking might occur due to transmission failure as well. In this case, some nodes (the number of $M \le N-1$) in WBN receive *block a* broadcast by node k successfully, while some nodes (the number of $N-M-1$) do not. We define the successful nodes and the node k as a set $\mathcal{M} = \{1,2,...,M, M+1\}$, and the failed nodes as a set $\mathcal{X} = \{1,2,...,N-M-1\}$. Consequentially, the nodes in $\mathcal{M}$ and that in $\mathcal{X}$ would work on the different block for mining.

If a node in $\mathcal{M}$ generates a valid block firstly, the nodes in $\mathcal{X}$ will give up the current mining process to work on this block, and the reason is that the block generated by $\mathcal{M}$ has the higher height to show the main chain in WBN clearly. In contrast, if a node x ($x \in \mathcal{X}$) generates a valid block, which is recorded as *block c*, before that in $\mathcal{M}$ (the node defined as node m, $m \in \mathcal{M}$, whose block is defined as *block d*), the unintentional forking would occur. The unintentional forking probability in *case* 2 can be defined as follows:

$$P\{case\ 2\} = P_r \cdot P\{T_k \le T_l, T_k + h \le T_l, T_x < T_m\}, \tag{12}$$

where T_x denotes the time when the node $x \in X$ generatesc *block c* , T_m denotes the time when the node $m \in M$ generates *block d*, and $P_r = C_{N-1}^{M} \cdot (P_c)^M \cdot (1-P_c)^{N-M-1}$ denotes the probability that M nodes successfully receive the broadcast of the node k and $N-M-1$ nodes have not.

Based on conditional probability, we have

$$P\{case\ 2\} = P_r \cdot P\{T_x < T_m \mid T_k \le T_l\} \cdot P\{T_k \le T_l\} \cdot P\{T_k + h \le T_l \mid T_k \le T_l\}. \quad (13)$$

T_l follows exponential distribution, based on the memoryless property of exponential distribution, we have $P\{T_k + h \le T_l \mid T_k \le T_l\} = P\{T_l \ge h\}$, and equation (13) can be rewritten as

$$P\{case\ 2\} = P_r \cdot P\{T_k \le T_l\} \cdot P\{T_l \ge h\} \cdot P\{T_x < T_m \mid T_k \le T_l\}. \quad (14)$$

Similar to the analysis of *case* 1 , the distribution of T_l , T_x and T_m is affected by the node which generates *block a*. Therefore, the unintentional forking probability in *case* 2 can be expressed as

$$P\{case\ 2\} = P_r \cdot \sum_{k=1}^{N} P\{T_k \le T_{-k}\} \cdot P\{T_{-k} \ge h\} \cdot P\{T_x < T_m \mid T_k \le T_{-k}\}. \quad (15)$$

In summary, the unintentional forking probability in *case* 2 is

$$P\{case\ 2\} = C_{N-1}^{M} \cdot (P_c)^M \cdot (1-P_c)^{N-M-1} \cdot \sum_{k=1}^{N} \frac{r_k}{\sum_{i=1}^{N} r_i} \cdot \exp\left(-\frac{\sum_{i=1}^{N} r_i - r_k}{D} h\right) \cdot \frac{\sum_{\forall X} r_j}{\sum_{i=1}^{N} r_i}. \quad (16)$$

5. Performance analysis

In the previous section, we investigated the causes of unintentional forking and derived the probability expression of unintentional forking in WBN. In this section, we discuss the impact of unintentional forking on system performance from three aspects: resource utilization rate, effective block generating time, and effective throughput[30].

Definition1 (Resource utilization rate). *The resource utilization rate in WBN is the ratio of the nodes working on the main chain to the total nodes.*

In order to simplify our analysis, we assume that the computational power of all nodes is $r_i = r$. Considering the forking probability, the expression of resource utilization rate (denotes as Λ) is

$$\Lambda = 1 - P + P\{case\ 1\} \cdot \sum_{i=1}^{N-1} \frac{2i^2}{N^2 \cdot (N-1)} + P\{case\ 2\} \cdot \frac{(M+1)^2 + (N-M-1)^2}{N^2}. \quad (17)$$

According to the introduction in Section Ⅲ, we can know that the time period T of the block is exponentially distributed, and the expression of the effective block generating time (denotes as $E[\Gamma]$) can be expressed as

$$E[\Gamma] = \frac{D}{N \cdot \Lambda \cdot r}. \quad (18)$$

Definition2 (Effective throughput). *Effective throughput is the number of transactions processed by the system per unit of time, which denotes as TPS.*

Due to the block size limitation, no matter how many new transactions arrive within the duration of effective block generating time $E[\Gamma]$, the maximum number of processed transactions cannot exceed the block size limitation L . In the case of communication

without delay and throughput constraints, TPS is determined by the maximum number of processed transactions L and new transaction arrival rate λ, which can be expressed as

$$TPS = \begin{cases} \sum_{i=1}^{N} \lambda_i, & \sum_{i=1}^{N} \lambda_i \le \frac{L}{E[\Gamma]} \\ \frac{L}{E[\Gamma]}, & \sum_{i=1}^{N} \lambda_i > \frac{L}{E[\Gamma]} \end{cases}, \quad (19)$$

where λ_i is the new transaction arrival rate of node i.

6. SIMULATION RESULTS AND DISCUSSIONS

In this section, we evaluate the unintentional forking probability of a PoW-based blockchain network considering the impact of transmission delay, target difficulty value, network scale, and transmission success probability. Moreover, we also investigate how the unintentional forking probability affects the performance of resource utilization, effective block generation time, and effective throughput. Without loss of generality, we assume that the number of nodes $N = 5$ in blockchain network (exclude the network scale experiment), the computational power of each miner $r = 1$, the target difficulty value $D = 100$, the broadcast delay $h = 10$, and the transmission success probability $P_c = 0.7$. Other parameters will be explained in the following experiments.

6.1. *The broadcast delay*

Fig. 2 shows the unintentional forking probability under different transmission delays, and some interesting observations can be obtained: 1) The unintentional forking probability in *case* 1 increases with the broadcast delay h, because the remaining nodes have more time to mine based on the previous block, so the chances of generating a valid block resulting in unintentional forking would be increased. 2) It is easy to see that as the broadcast delay goes up, the unintentional forking probability in *case* 2 decreases. This is because it has happened in *case* 1 yet. 3) By comparing the results of *case* 1, *case* 2 and the entire consensus process (*case* 1+*case* 2), we can see that the unintentional forking in WBN mainly occurs in *case* 1, which is caused by transmission delay.

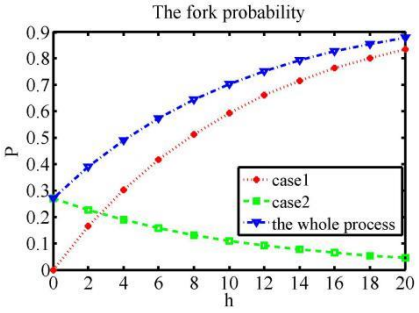


Fig. 2: The fork probability P with different h

6.2. *The target difficulty*

In the second experiment, we change the target difficulty value (D) to show its impact on the unintentional forking probability. As shown in Fig. 3, it can be seen that increasing target difficulty value can improve the phenomenon of unintentional forking in WBN. This is because the block generation speed is related to the target difficulty value and the computing power of node. Therefore, increasing the target difficulty value can slow down the block generation speed to reduce the probability of orphaning a block generating the unintentional forking.

6.3. *Network scale*

Next, we change the number of nodes to observe the effect of network scale on the unintentional forking probability in WBN. From Fig. 4, it can be easily seen that the

unintentional forking probability in *case* 1 increases with N, because the more nodes there are, the greater the computing power remaining nodes to generate a new valid block for forking. In contrast, since the network scale cannot affect the transmission failure, and the unintentional forking occurs in *case* 1 mainly, we can notice that the unintentional forking probability in *case* 2 increases with the increase of N. Additionally, the total unintentional forking probability increases with the network scale, and thus it is necessary to control the block generation speed in a large network scale.

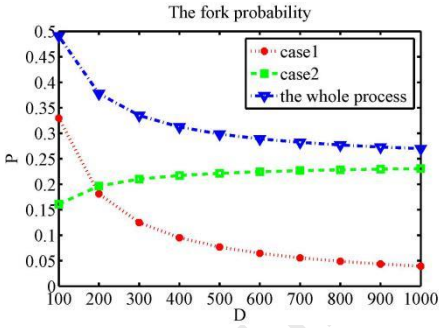


Fig. 3: The fork probability P with different D

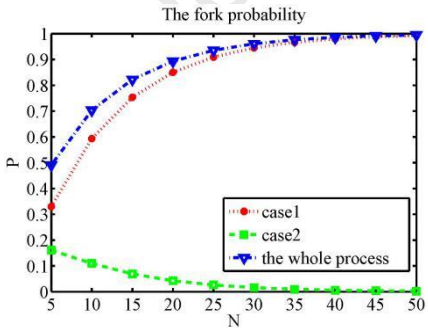


Fig. 4: The fork probability P with different N

6.4. *Probability of transmission success*

Fig. 5 depicts that the unintentional forking probability changes transmission success probability. Intuitively, we can see that increasing transmission success probability can decrease the unintentional forking probability, because it can reduce the probability in *case* 2.

For *case* 2, the occurrence of the unintentional forking is the computational power competition between the nodes that successfully receive the broadcast and the nodes do not. The larger P_c indicates the smaller number of unsuccessful nodes, and the corresponding computational power would be less to generate the forking.

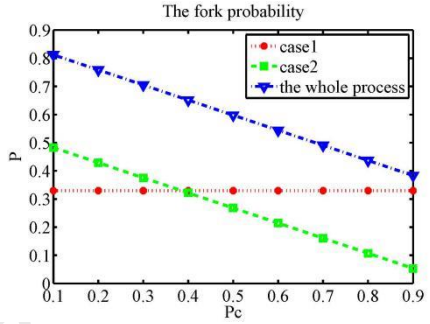


Fig. 5: The fork probability P with different P_c

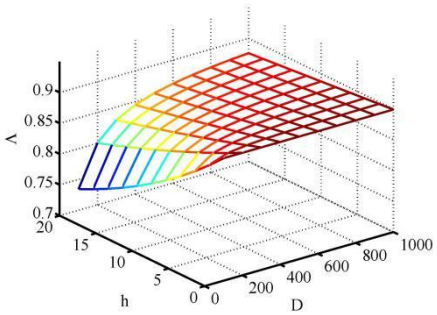


Fig. 6: The effective resource utilization rate Λ

6.5. *System performance*

In this part, the impact of unintentional forking on resource utilization rate, effective block generation time and effective throughput are investigated, and the simulation results are illustrated in Fig. 6, Fig. 7, and Fig. 8, respectively.

In this experiment, let $P_c = 0.7$, $N = 5$, we change D and h to compare resource utilization rates under different unintentional forking probabilities. The results in Fig. 6 show that the unintentional forking will reduce

system resource utilization rate. This is because that the unintentional forking would waste a lot of computational power to generate orphan blocks, thereby reducing the computational power consumption on the main chain. In addition, the simulation results also reflect the relationship between the unintentional forking probability and target difficulty value, as well as broadcast delay.

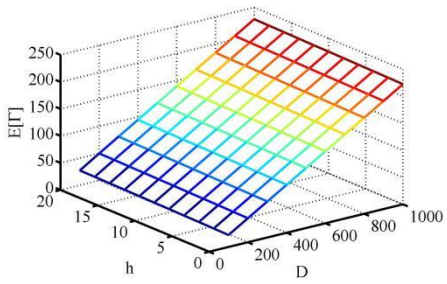


Fig. 7: The effective time to generate block $E[\Gamma]$

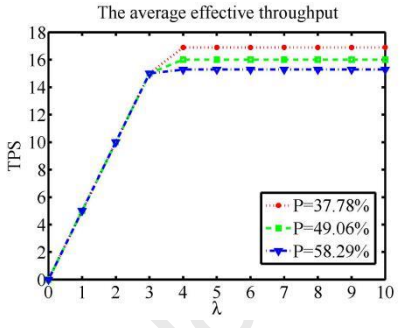


Fig. 8: The average effective throughput TPS with different λ

Fig. 7 discusses the effective block generation time for different unintentional forking probabilities. Specifically, when the broadcast delay increases, effective block generation time increases. This is because the increase of broadcast delay can increase the unintentional forking probability, and the block generation speed would be reduced due to the less computational power consumption on the main chain.

The last experiment is to evaluate the impact of unintentional forking on effective throughput (TPS). We set the maximum number of processed transactions $L = 400$ in each block, and change new transaction arrival rate to compare the effective throughput under different unintentional forking probabilities with $P = 37.78\%$, $P = 49.06\%$, $P = 58.29\%$, respectively[1]. The results in Fig. 8 show that the effective throughput increases with the arrival rate of new transaction and eventually converges to a stable value. Moreover, for a higher unintentional forking probability, effective throughput will become smaller, because this will reduce the new block generation speed. Note that when increasing target difficulty value to reduce the unintentional forking probability, we need to consider the trade-off between resource utilization rate and effective throughput in WBN, in which a larger target difficulty value will improve the system's resource utilization but reduce the effective throughput.

7. CONCLUSIONS

In this paper, we analyze and discuss the unintentional forking in WBN. We analyze the case of unintentional forking at the same blockchain height, and derive the unintentional forking probability expressions in each case. In addition, we also analyze the factors that affect unintentional forking, and evaluate the impact of unintentional forking on main performances such as resource utilization, average time to generate new blocks, and TPS. Through this work, it is easy to know which factors in the network will affect the unintentional forking probability, and we can effectively avoid unintentional forking in WBN from these analyses.

Different from the research of malicious

[1] the corresponding P is computed based on equations (11) and (16) letting $D = 10$ with $h = 5,10,15$.

attacks, this paper clearly indicates that communication transmission plays an important role in the performance and security of the blockchain system without any attacker. Therefore, in order to design an efficient and secure blockchain system, a reasonable solution is to make a balance between communication transmission and consensus protocols. Future work may consider the unintentional forking process and its effect with multi-forking at different heights, and study the design of optimization algorithm or protocol to reduce the unintentional forking probability in WBN.

References

[1] Z. Yu, X. Liu and G. Wang, A Survey of Consensus and Incentive Mechanism in Blockchain Derived from P2P, in: Parallel and Distributed Systems (ICPADS), 2018 24th International Conference on, IEEE, 2018, pp. 1010-1015.

[2] B. Cao, L. Zhang, Y. Li, D. Feng and W. Cao, Intelligent Offloading in Multi-Access Edge Computing: A State-of-the-Art Review and Framework, IEEE Commun Mag. 57 (3) (Mar. 2019) 56-62.

[3] B. Cao, S. Xia, J. Han and Y. Li, A Distributed Game Methodology for Crowdsensing in Uncertain Wireless Scenario, IEEE Trans. Mob. Comput. 19 (1) (Jan. 2020) 15-28.

[4] Y. Li, S. Xia, Q. Yang, G. Wang and W. Zhang, Life Priority Driven Resource Allocation for WNV-Based Internet of Things, IEEE. IoT. J. (2020) 1-1.

[5] L. Huang, X. Feng, C. Zhang, L.Qian and Y. Wu, Deep reinforcement learning-based joint task offloading and bandwidth allocation for multi-user mobile edge computing, Digital Commun. Netw. 5 (1) (2019) 10-17.

[6] Y. Ai, M. Peng and K. Zhang, Edge computing technologies for Internet of Things: a primer, Digital Commun. Netw. 4 (2) (2018) 77-86.

[7] L. Zhang, B. Cao, Y. Li, M. Peng and G. Fang, A Multi-Stage Stochastic Programming based Offloading Policy for Fog Enabled IoT-eHealth, accepted by IEEE Journal on Selected Areas in Communications, Apr. 2020.

[8] H. Dai, Z. Zheng and Y. Zhang, Blockchain for Internet of Things: A Survey, IEEE. IoT. J. 6 (5) (Oct. 2019) 8076-8094.

[9] B. Lucas and R. V. Páez, Consensus Algorithm for a Private Blockchain, in: Electronics Information and Emergency Communication (ICEIEC), 2019 9th International Conference on, IEEE, 2019, pp. 264-271.

[10] C. T. Nguyen, D. T. Hoang, D. N. Nguyen, D. Niyato, H. T. Nguyen and E. Dutkiewicz, Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities, IEEE Access, 7 (2019) 85727-85745.

[11] H. Xu, L. Zhang, Y. Liu and B. Cao, RAFT Based Wireless Blockchain Networks in the Presence of Malicious Jamming, IEEE Wireless Communications Letters, (Feb. 2020) 1-1.

[12] Castro M and Liskov B, Practical Byzantine fault tolerance, in: Proceedings of the third symposium on Operating systems design and implementation (OSDI), USENIX Association, USA, 1999, pp. 173-186.

[13] S. Wang, C. Wang and Q. Hu, Corking by Forking: Vulnerability Analysis of Blockchain, in: 2019 IEEE Conference on Computer Communications, IEEE INFOCOM, 2019, pp. 829-837.

[14] B. Liu, Y. Qin and X. Chu, Reducing Forks in the Blockchain via Probabilistic Verification, in: Data Engineering Workshops (ICDEW), 2019 35th International Conference on, IEEE, 2019, pp. 13-18.

[15] F. J. Couto da Silva, S. B. Damsgaard, M. A. M. Sørensen, F. Marty, B. Altariqi, E. Chatzigianni, T. K. Madsen and H. P. Schwefel, Analysis of Blockchain Forking on an Ethereum Network, in: 2019 25th European Wireless Conference, 2019, pp. 1-6.

[16] S. Cheng and S. Lin, Mining Strategies for Completing the Longest Blockchain, IEEE Access, 7 (2019) 173935-173943.

[17] E. Anceaume, T. Lajoie-Mazenc, R. Ludinard and B. Sericola, Safety analysis of Bitcoin improvement proposals, in: Network Computing and Applications (NCA), 2016 15th International Symposium on, IEEE, 2016, pp. 318-325.

[18] M. Memon, U. A. Bajwa, A. Ikhlas, Y. Memon, S. Memon and M. Malani, Blockchain Beyond Bitcoin: Block Maturity Level Consensus Protocol, in: Engineering Technologies and Applied Sciences (ICETAS), 2018 5th International Conference on, IEEE, 2018, pp. 1-5.

[19] B. Cao, Y. Li, L. Zhang, L. Zhang, S. Mumtaz, Z. Zhou and M. Peng, When Internet of Things Meets Blockchain: Challenges in Distributed Consensus, IEEE Network, 33 (6) (Nov.-Dec. 2019) 133-139.

[20] G. BitFury, Proof of stake versus proof of work, White paper, Sep. 2015.

[21] J. G. Andrews, F. Baccelli and R. K. Ganti, A Tractable Approach to Coverage and Rate in Cellular Networks, IEEE Trans. Commun. 59 (11) (2011) 3122-3134.

[22] Y. Sun, L. Zhang, G. Feng, B. Yang, B. Cao and M. A. Imran, Blockchain-Enabled Wireless Internet of Things: Performance Analysis and Optimal Communication Node Deployment, IEEE. IoT. J. 6 (3) (2019) 5791-5802.

[23] Y. Li, B. Cao, M. Peng, L. Zhang, D. Feng and J. Yu, Direct Acyclic Graph- based Ledger for Internet of Things: Performance and Security Analysis, accepted by IEEE/ACM Transactions on Networking, Apr. 2020.

[24] Gervais A , Karame G O , Karl W\"{u}st, Glykantzis V, Ritzdorf H and Capkun S, On the Security and Performance of Proof of Work Blockchains, in: 2016 ACM SIGSAC Conference, ACM, 2016, pp. 3-16.

[25] Sapirshtein A, Sompolinsky Y and Zohar A, Optimal Selfish Mining Strategies in Bitcoin, in: 2016 International Conference on Financial Cryptography and Data Security, Springer, 2016, pp. 515-532.

[26] Heilman E, Kendler A, Zohar A and Goldberg S, Eclipse attacks on Bitcoin's peer-to-peer network, in: Proceedings of the 24th USENIX Conference on Security Symposium, USENIX Association, 2015, pp. 129-144.

[27] V. B. Mišić, J. Mišić and X. Chang, On Forks and Fork Characteristics in a Bitcoin-Like Distribution Network, in: 2019 IEEE International Conference on Blockchain (Blockchain), IEEE, 2019, pp. 212-219.

[28] C. Decker and R. Wattenhofer, Information propagation in the Bitcoin network, Proc. IEEE. P2P. (2013) 1-10.

[29] Y. Shahsavari, K. Zhang and C. Talhi, A Theoretical Model for Fork Analysis in the Bitcoin Network, in: 2019 IEEE International Conference on Blockchain (Blockchain), IEEE, 2019, pp. 237-244.

[30] B. Cao, Z. Zhang, D. Feng, S. Zhang, L. Zhang, M. Peng and Y. Li, Performance analysis and comparison of PoW, PoS and DAG based blockchains, accepted by Digital Communications and Networks, Jan. 2020.

Conflict of Interest

1. All authors of this research paper have direct Ly participated in the planning execution, or analysis of this study.

2. All authors of this paper have read and approved the final version submitted.

3. The contents of this manuscript have not been copyrighted or published previously.

4. There are no directly related manuscript or abstracts,pulished or unplished, by any authors of this paper.

5. All authors of this research paper have no conflict of interest, financial or otherwise.

Authors: Qilie Liu, Yinyi Xu, Bin Cao

Lei Zhang, Mugen Peng