UDC 004.75

Number of Confirmation Blocks for Bitcoin and GHOST Consensus Protocols on Networks with Delayed Message Delivery

L. V. Kovalchuk[1,3, a], D. S. Kaidalov[3], A. O. Nastenko[3], O. V. Shevtsov[2,3], M. Yu. Rodinko[2,3], R. V. Oliynykov[2,3, b]

[1] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"*
[2] *V. N. Karazin Kharkiv National University*
[3] *Input Output HK*

Abstract

A specific number of transaction confirmation blocks determines average time of receiving and accepting payments at cryptocurrencies, and the shortest confirmation time for the same level of blockchain security provides the best user properties. Existing papers on transaction confirmation blocks for Bitcoin use implicit assumption of prompt spreading of Bitcoin blocks over the network (that is not always the case for the real world conditions). The newer publications with rigorous analysis and proofs of Bitcoin blockchain properties that take into account network delays provide asymptotic estimates, with no specific numbers for transaction confirmation blocks.
We propose three methods for determination of required number of confirmation blocks for Bitcoin and GHOST on networks with delayed message delivery with different models that take into account the possibility of faster adversarial node syncronization. For the GHOST we propose the first (to our knowledge) strict theoretical method that allows to get required number of confirmation blocks for a given attacker's hashrate and attack success probability.

Keywords: Bitcoin, GHOST, consensus protocol, Proof-of-Work

Introduction

Bitcoin and many other altcoins provide decentralized payment services with no trusted parties. Modern cryptocurrencies are based on public transaction ledgers (blockchains) that are maintained by each participant (a full node) of a distributed peer-to-peer network. Consistent transaction ledger is built using consensus protocol that must be robust to arbitrary behavior of an attacker with bounded resources, as well as to honest nodes' failures or network outages. The latter leads to the possibility of existing several unintentional alternative histories of blockchain concurrently run by honest nodes, and ability of consensus protocol to select the only one "correct" version of blockchain among several available branches on discovering them.

These properties of cryptocurrency distributed consensus protocols also allow intentional adversarial creation of a blockchain branch for a *double spend* attack, when a transaction is reverted or cancelled (e.g., after a merchant sent goods or provided services), so an attacker gets goods or services and finally keeps his coins back.

To prevent such type of attacks (to decrease their success probability to acceptable small threshold), it is necessary to wait for some amount of blocks that follow the one with the transaction of interest, after which it is accepted by merchant.

The exact number of such confirmation blocks is important for application properties of cryptocurrency and closely related to average time of receiving and accepting payments. The shortest confirmation time for the same level of transaction security provides the best user properties for cryptocurrency.

Previous work. The first model that shows exponential decreasing of attack probability success with number of confirmation blocks was shown in the original Bitcoin paper [1]. It uses a random walk process with a single random variable that follows binomial distribution (with Poisson approximation). There is also an implicit assumption of prompt spreading of Bitcoin blocks over a peer-to-peer network. Though several honest chains were mentioned that may be visible to nodes in the paper, the model takes into account only an intentionally built alternative adversarial chain.

The paper of M.Rosenfeld [2] uses an assumption of a random variable that follows a negative binomial distribution for defining of the difference in the number of blocks generated by honest miners and by an adversary. The later paper [3] by C.Grunspan and R.Perez-Marco provides proofs on selection of the negative binomial distribution of the analyzed random variable, and gives strict estimates of the number of confirmation blocks. The paper of C.Pinzon and C.Rocha [4] generalizes the approach from [2] and incorporates generation time to the model of double spend attack.

All mentioned papers use models with implicit assumption of prompt spreading of Bitcoin blocks over the network that leads to following consequences:

[a] lusi.kovalchuk@gmail.com
[b] roman.oliynykov@iohk.io

- network synchronization is done promptly, and each block is visible to all nodes immediately at the very moment it is mined and published;
- two or more honest miners cannot generate blocks simultaneously (probability of this event is equal to zero), as well as it is impossible to create an unintentional fork;
- probability of existence of two different chains having the same length mined by honest miners is also equal to zero;
- speed of the growing main chain is equal to honest miners' block generation speed.

These statements are not always the case for the real world conditions of cryptocurrencies application, so a different model should be used that should take into account delays introduced by peer-to-peer network message delivery.

The paper [5] introduces a formal definition and analysis of Bitcoin backbone protocol when the participants operate in a synchronous and partially synchronous communication network (that has an upper bound for delays of message delivery). An approach for formal analysis in asynchronous networks was presented at [6]. Further development of [5] is presented at [7] that allows strict formalization of target recalculation function in Bitcoin.

These papers provide generalized analysis with proofs of asymptotic estimates on achievement of main blockchain properties (persistence and liveness), but do not give any method for computation of the required number of confirmation blocks for cryptocurrency practical application.

In [8] and [9] a tradeoff on transaction throughput and security of blockchains were studied, and the GHOST rule was proposed that allows achieving higher transaction rates via adoption of tree data structures for keeping blocks. A discussion of options for some proofs was presented. E.g., Proposition 11 at [8]: from inequality 1 in the proof it follows that the rate of block addition to the main chain by honest miners only $\beta(\lambda_h)$ is higher than the rate of block addition when main chain is extended both by honest users blocks and a fraction f of the attacker's blocks: $\beta(\lambda_h) \geq \beta(\lambda_h + f \cdot q \cdot \lambda_h)$; monotonically decreasing properties of the $\beta(\lambda)$ function on its argument follow from the same inequality (i.e., with increase of the speed of block generation λ, the rate of block addition to the main chain is decreased). These papers also provide upper and lower bounds of the rate of block addition to the main chain, but there is no published strict theoretical method (to our knowledge) for computation of the required number of confirmation blocks in cryptocurrencies that utilize GHOST.

Our results. Within a model of a synchronous communication network with limited delays of message delivery [5, 10], we develop several methods for determination of the required number of confirmation blocks for Bitcoin and GHOST. The first model considers equal delays for message delivery on the Bitcoin peer-to-peer network both for honest and malicious miners. The second model for Bitcoin assumes that an attacker may create his own centralized network with faster synchronization, thus optimizing attack efficiency. The last model is for GHOST and takes into account its tree data structure for organizing of blocks, the longest chain selection rule and much shorter time between blocks. For each model we develop a method for determination of the required number of confirmation blocks with a given attacker's hashrate and attack success probability.

1. Notations and auxiliary statements

We define a timeslot (TS) as the period of synchronization, i.e. the amount of time needed to share a block between independent miners. We introduce a value s_H which is the ratio $\frac{t_1}{t_2}$, where t_1 is the period of network synchronization for honest miners and t_2 is the time needed for one attempt of block generation (roughly speaking, time of random oracle of hash function request processing). It means that each honest miner (HM) can make approximately s_H attempts to generate a block, before he can see a block generated by some other HM in this TS. For a malicious miner (MM), we assume $s_M = s_H$ for the first model and $s_M = \frac{s_H}{2}$ for the second one. For the third model, we assume $s_M = s_H = s$.

We also use the following notations and assumptions:

- p is the probability to generate a block by one miner in one attempt; roughly speaking, this is the probability to generate an appropriate pre-image of some hash-function (we assume $p = \frac{1}{k \cdot n \cdot s_H}$, where k is the ratio of block generation time to network block propagation time);

- n is the number of HMs;

- m is the number of MMs (we assume that $m < n$, so honest miners have majority).

Also we emphasize once more that in Model 1 HMs and MMs can extend the blockchain not more than by one block during one TS, in Model 2 HMs can extend the blockchain not more than one block during one TS, but MMs, using their advantage in synchronization time, can extend it by one or two blocks during one TS. In Model 3, HMs can extend the blockchain not more than by three blocks during one TS and MMs can extend the blockchain not more than by two blocks during one TS.

Now we need to define and to calculate some probabilities that we will use in further statements.

In Models 1 and 2 for HMs the probability to generate nothing during one TS is

$$p_0 = (1-p)^{n \cdot s_H},$$

and the probability to extend the blockchain exactly by one block is

$$p_1 = 1 - p_0.$$

For MMs, the probability to generate nothing during one TS is

$$q_0 = (1-p)^{m \cdot s_H},$$

the probability to extend the blockchain exactly by two blocks is

$$q_2 = \left(1 - (1-p)^{m \cdot s_m}\right)^2,$$

and the probability to extend the blockchain exactly by one block is

$$q_1 = 1 - q_0 - q_2.$$

Note that for the Model I: $q_2 = 0$.

Also, for Model 3 we introduce the corresponding probabilities:

$$p_i = C_{ns}^i p^i (1-p)^{ns-i},\ i = 0, 1, 2; \tag{1}$$

$$p_3 = 1 - p_0 - p_1 - p_2;$$

and

$$q_i = C_{ms}^i p^i (1-p)^{ms-i},\ i = 0, 1,$$
$$q_2 = 1 - q_0 - q_1, \tag{2}$$

where s is the number of attempts in one TS (for Model 3, the parameter s is the same that S_H for Models 1 and 2).

To prove our main result, we need auxiliary lemmas. The first and the second ones are some kind of ruin problem generalizations. We formulate them in this section. The others will be formulated in sections 4 and 5. To formulate the lemmas, we introduce some additional notations.

Let $\{\xi_i, i \geq 1\}$, and $\{\eta_i, i \geq 1\}$ be mutually independent random variables (RVs), where for all $i \geq 1$

$$\xi_i = \begin{cases} 0, & \text{with probability } p_0; \\ 1, & \text{with probability } p_1, \end{cases} \tag{3}$$

$$\eta_i = \begin{cases} 0, & \text{with probability } q_0; \\ 1, & \text{with probability } q_1; \\ 2, & \text{with probability } q_2, \end{cases} \tag{4}$$

and define RVs $\{\zeta_i, i \geq 1\}$, as

$$\zeta_i = \xi_i - \eta_i.$$

It is easy to calculate probability distribution of ζ_i, $i \geq 1$:

$$P_0 := P(\zeta_i = 0) = p_0 q_0 + p_1 q_1;$$

$$P_1 := P(\zeta_i = 1) = p_1 q_0;$$

$$P_{-1} := P(\zeta_i = -1) = p_0 q_1 + p_1 q_2;$$

$$P_{-2} := P(\zeta_i = -2) = p_0 q_2.$$

Also let us define RVs as

$$S_n = \sum_{i=1}^{n} \xi_i, n \geq 1; S_0 = 0;$$

$$\Sigma_n = \sum_{i=1}^{n} \eta_i - k,\ n \geq 1;\ \Sigma_0 = -k \text{ for some } k \in N$$

and

$$L_n = S_n - \Sigma_n, n \geq 1; L_0 = k.$$

We can also write L_n as $L_n = \sum_{i=1}^{n} \zeta_i + k$. From the probability distribution of ζ_i, we get the following equalities:

$$L_{n+1} = \begin{cases} L_n - 2, & \text{with prob. } P_{-2}; \\ L_n - 1, & \text{with prob. } P_{-1}; \\ L_n, & \text{with prob. } P_0; \\ L_n + 1, & \text{with prob. } P_1. \end{cases} \tag{5}$$

Now we are ready to formulate the first lemma.

Lemma 1. *Define the event A_k as*
$A_k = \{\exists\, n \geq 1 : \quad L_n \leq 0\}$ *and let $q^{(k)} = P(A_k)$. Then if the condition*

$$P_{-1} + 2P_{-2} < P_1 \tag{6}$$

holds, then

$$q^{(k)} = \frac{(1-\lambda_2)\lambda_1^{k+1} - (1-\lambda_1)\lambda_2^{k+1}}{\lambda_1 - \lambda_2}, \tag{7}$$

where

$$\lambda_1 = \frac{P_{-1} + P_{-2} - \sqrt{(P_{-1} + P_{-2})^2 + 4P_{-1}P_{-2}}}{2P_1},$$

$$\lambda_2 = \frac{P_{-1} + P_{-2} + \sqrt{(P_{-1} + P_{-2})^2 + 4P_{-1}P_{-2}}}{2P_1}.$$

Proof. To prove the Lemma, we will derive a differential equation for $q^{(k)}$ using (5) and solve it.

According to the compound probability formula

$$q^{(k)} = P(A_k) = P\left(A_k/\zeta_1 = -2\right) P_{-2} +$$

$$+ P\left(A_k/\zeta_1 = -1\right) P_{-1} +$$

$$+ P\left(A_k/\zeta_1 = 0\right) P_0 + P\left(A_k/\zeta_1 = 1\right) P_1 =$$
$$= q^{(k-2)} P_{-2} + q^{(k-1)} P_{-1} + q^{(k)} P_0 + q^{(k+1)} P_1,$$

where the second equality is due to (5). We can rewrite it as

$$q^{(k-2)} P_{-2} + q^{(k-1)} P_{-1} +$$

$$+ q^{(k)} (P_0 - 1) + q^{(k+1)} P_1 = 0. \tag{8}$$

The corresponding characteristic equation is

$$\lambda^3 P_1 + \lambda^2 (P_0 - 1) + \lambda P_{-1} + P_{-2} = 0$$

with one obvious root $\lambda = 1$. After division by $\lambda - 1$, we obtain a new equation:

$$\lambda^2 P_1 - \lambda\left(P_{-1} + P_{-2}\right) - P_{-2} = 0.$$

Its discriminant is positive:

$$\left(P_{-1} + P_{-2}\right)^2 + 4P_{-1}P_{-2} > 0,$$

so the equation has two real roots:

$$\lambda_1 = \frac{P_{-1} + P_{-2} - \sqrt{\left(P_{-1} + P_{-2}\right)^2 + 4P_{-1}P_{-2}}}{2P_1},$$

$$\lambda_2 = \frac{P_{-1} + P_{-2} + \sqrt{\left(P_{-1} + P_{-2}\right)^2 + 4P_{-1}P_{-2}}}{2P_1}.$$

Also we can see that $\lambda_1 < 0$ because of

$$P_{-1} + P_{-2} = \sqrt{\left(P_{-1} + P_{-2}\right)} <$$

$$< \sqrt{\left(P_{-1} + P_{-2}\right) + 4P_{-1}P_{-2}}$$

and $\lambda_1 > -1$ because of

$$P_1 + P_{-1} > 0.$$

The general solution of (8) is

$$q^{(k)} = a_1\lambda_1^k + a_2\lambda_2^k,$$

where a_1 and a_2 can be found from the boundary conditions

$$q^{(0)} = q^{(-1)} = 1. \tag{9}$$

The boundary conditions (9) lead to

$$\begin{cases} a_1 + a_2 = 1; \\ a_1\lambda_1 + a_2\lambda_2 = \lambda_1\lambda_2, \end{cases}$$

whence we obtain

$$a_1 = \frac{\lambda_1\left(1 - \lambda_2\right)}{\lambda_1 - \lambda_2}; a_2 = \frac{\lambda_2\left(1 - \lambda_1\right)}{\lambda_1 - \lambda_2}$$

and, finally,

$$q^{(k)} = \frac{\left(1 - \lambda_2\right)\lambda_1^{k+1} - \left(1 - \lambda_1\right)\lambda_2^{k+1}}{\lambda_1 - \lambda_2}.$$

But $q^{(k)}$ is the probability of some event, so we should guarantee that it is not smaller than 0 and is not larger than 1.

The inequality $q^{(k)} > 0$ implies from the facts that $1 - \lambda_2 < 1 - \lambda_1$, $\lambda_1^k < \lambda_2^k$ (because of $|\lambda_1| < |\lambda_2|$, λ_1 is negative, λ_2 is positive) and $\lambda_1 - \lambda_2 < 0$.

Now we will prove that the inequality $q^{(k)} < 1$ follows from the condition $P_{-1} + 2P_{-2} < P_1$ of this lemma. Note that the condition $\lambda_2 < 1$ is sufficient for $q^{(k)} < 1$. Thus, if $\lambda_2 < 1$ then we obtain

$$q^{(k)} = \frac{\left(1 - \lambda_2\right)\lambda_1^{k+1} - \left(1 - \lambda_1\right)\lambda_2^{k+1}}{\lambda_1 - \lambda_2} <$$

$$< \frac{\left(1 - \lambda_2\right)\lambda_2^{k+1} - \left(1 - \lambda_1\right)\lambda_2^{k+1}}{\lambda_1 - \lambda_2} =$$

$$= \frac{\left(1 - \lambda_2\right) - \left(1 - \lambda_1\right)}{\lambda_1 - \lambda_2}\lambda_2^{k+1} = \lambda_2^{k+1} < 1.$$

Now we have only to prove that the condition $P_{-1} + 2P_{-2} < P_1$ involves the condition $\lambda_2 < 1$. The former inequality holds iff

$$P_{-1} + P_{-2} + \sqrt{\left(P_{-1} + P_{-2}\right)^2 + 4P_{-1}P_{-2}} < 2P_1,$$

or iff

$$\sqrt{\left(P_{-1} + P_{-2}\right)^2 + 4P_{-1}P_{-2}} < 2P_1 - P_{-1} - P_{-2},$$

or iff

$$\begin{cases} P_{-1} + P_{-2} < 2P_1; \\ \left(P_{-1} + P_{-2}\right)^2 + 4P_1P_2 < \left(2P_1 - P_{-1} - P_{-2}\right)^2. \end{cases}$$

Direct calculations show that the latter system is equivalent to the inequality $P_{-1} + 2P_{-2} < P_1$.

The Lemma is proved.

□

Corollary 1. *In the particular case when $q_2 = 0$ we obtain*

$$q^{(k)} = \left(\frac{p_0q_1}{p_1q_0}\right)^k.$$

Proof. In the case of $q_2 = 0$, we get the following equalities:

$$P_{-2} = 0;\ \lambda_1 = 0;\ \lambda_2 = \frac{p_0q_1}{p_1q_0};\ a_2 = 1.$$

Then $q_k = \lambda_2^k = \left(\frac{p_0q_1}{p_1q_0}\right)^k$. □

We also need a more complicated lemma that will be proved using Lemma 1. Let $\{\nu_i, i \geq 1\}$ be independent identically distributed RV, which are also mutually independent with $\{\eta_i, i \geq 1\}$, introduced in (4). Their probability distribution is

$$\nu_i = \begin{cases} 0, & \text{with probability } r_0; \\ 1, & \text{with probability } r_1; \\ 2, & \text{with probability } r_2; \\ 3, & \text{with probability } r_3. \end{cases} \tag{10}$$

We are going to formulate some statement for RVs (4) and (9), which is more general than Lemma 1, formulated for RVs (3) and (4).

Let us define RV $\{\gamma_i, i \geq 1\}$ as

$$\gamma_i = \nu_i - \eta_i.$$

It is easy to prove that for all $i \geq 1$:

$$\begin{aligned} R_0 &:= P(\gamma_i = 0) = r_0 q_0 + p_1 q_1 + p_2 q_2; \\ R_1 &:= P(\gamma_i = 1) = r_1 q_0 + r_2 q_1 + r_3 q_2; \\ R_2 &:= P(\gamma_i = 2) = r_2 q_0 + r_3 q_1; \\ R_3 &:= P(\gamma_i = 3) = r_3 q_0; \\ R_{-1} &:= P(\gamma_i = -1) = r_0 q_1 + r_1 q_2; \\ R_{-2} &:= P(\gamma_i = -2) = r_0 q_2. \end{aligned} \tag{11}$$

Also define RVs $U_n = \sum_{i=1}^{n} \nu_i,\ n \geq 1,\ U_0 = 0$, and

$$T_n = U_n - \Sigma_n,\ n \geq 1,\ T_0 = k.$$

Note that $T_n = \sum_{i=1}^{n} \gamma_i + k,\ n \geq 1$.

From (11) we obtain that

$$T_n = \begin{cases} T_{n-1} - 2, & \text{with probability } R_{-2}; \\ T_{n-1} - 1, & \text{with probability } R_{-1}; \\ T_{n-1}, & \text{with probability } R_0; \\ T_{n-1} + 1, & \text{with probability } R_1; \\ T_{n-1} + 2, & \text{with probability } R_2; \\ T_{n-1} + 3, & \text{with probability } R_3. \end{cases}$$

Lemma 2. *Let us define the event*

$$B_k = \{\exists\ n \geq 1 : T_n \leq 0\}.$$

Also, define $Q_1 = R_1 + R_2 + R_3$.

Then if the condition

$$R_{-1} + 2R_{-2} < Q_1 \tag{12}$$

holds, then $P(B_k) \leq Q^{(k)}$, *where*

$$Q^{(k)} = \frac{(1-\lambda_2)\lambda_1^{k+1} - (1-\lambda_1)\lambda_2^{k+1}}{\lambda_1 - \lambda_2},$$

$$\lambda_1 = \frac{R_{-1} + R_{-2} - \sqrt{(R_{-1} + R_{-2})^2 + 4R_{-1}R_{-2}}}{2Q_1},$$

$$\lambda_2 = \frac{R_{-1} + R_{-2} + \sqrt{(R_{-1} + R_{-2})^2 + 4R_{-1}R_{-2}}}{2Q_1}.$$

Proof. Let us introduce new RVs $\{\delta_i, i \geq 1\}$ that are obtained from ν_i in such a way:

$$\delta_i = \begin{cases} \nu_i,\ if\ \nu_i \in \{0,1\}; \\ 1,\ if\ \nu_i \in \{2,3\}; \end{cases} \tag{13}$$

It is easy to see that $\forall i \geq 1 : \delta_i \leq \nu_i$, and therefore ,

$$Z_n = \sum_{i=1}^{n} \delta_i \leq U_n, n \geq 1;$$

$$Y_n = Z_n - \Sigma_n + k \leq T_n,\ n \geq 1. \tag{14}$$

Let us introduce the event

$$C_k = \{\exists\ n \geq 1 : Y_n \leq 0\}.$$

From the definition of B_k and (14) we get that $B_k \subset C_k$ and

$$P(B_k) \leq P(C_k). \tag{15}$$

Next, from (13) we get that

$$\delta_i = \begin{cases} 0, \text{ with probability } R_0, \\ 1, \text{ with probability } \\ \quad Q_1 = R_1 + R_2 + R_3. \end{cases} \tag{16}$$

Then we can apply Lemma 1 to RVs (4) and (13) and obtain the probability $P(C_k) = Q^{(k)}$, and then use inequality (15) to complete the proof of this Lemma. □

2. Model 1. Fork probability for an adversary with ordinary synchronization.

Let us fix some $N \in \mathbb{N}$ and consider a part of blockchain from TS number $t_0 = 1$ to TS number N.

We define the event:

$F(l, N) = \{$ the fork occurred, that started at $t_0 = 1$ and got the length l before the TS number N, under the condition that HMs generated l confirmation blocks starting at t_0 $\}$.

Theorem 1. *For the event* $F(l, N)$, *the following upper bound holds:*

$$\begin{aligned} P(F(l,N)) \leq \sum_{l_0=0}^{N-l} \Bigg[& C_{l+l_0-1}^{l-1} p_1^l (1-p_1)^{l_0} \cdot \\ & \cdot \Bigg((1 - \sum_{k=0}^{l-1} C_{l+l_0}^{k} q_1^k \times (1-q_1)^{l+l_0-k}) + \\ & + \sum_{k=0}^{l-1} \Bigg\{ C_{l+l_0}^{k} q_1^k (1-q_1)^{l+l_0-k} \cdot \\ & \cdot \left(\frac{q_1(1-p_1)}{p_1(1-q_1)} \right)^{l-k} \Bigg\} \Bigg) \Bigg]. \end{aligned} \tag{17}$$

Proof. It is obvious that $F(l, N) \subset \cup_{l_0=0}^{N-l} F_{l,l_0}$,

where F_{l,l_0} is the event

$F_{l,l_0} = \{$ the fork occurred after HMs generated l confirmation blocks, and they generated these blocks exactly during $l + l_0$ TSs starting from $t_0 = 1\}$.

Also for some fixed $l, l_0 \in \mathbb{N}$ we introduce the following events:

$H_{l,l_0} = \{$ HMs generated l confirmation blocks during exactly $l + l_0$ TSs, starting at $t_0 = 1\}$;

$M = \{$ MMs generated not less then l (i.e. l or more) blocks during exactly $l + l_0$ TSs, starting at t_0 $\}$;

$M_k = \{$ MMs generated exactly k $(0 \leq k \leq l-1)$ blocks during $l + l_0$ TSs, starting at t_0 $\}$;

$H_{l-k}^{\infty} = \{$ MMs ever catch up with the honest chain under the condition that in TS $l + l_0$ they are exactly $l - k$ blocks behind $\}$.

From the definition of F_{l,l_0}, we get

$$F_{l,l_0} \subset H_{l,l_0} \cap (M \cup (\cup_{k=0}^{l-1}(M_k \cap M_{l-k}^{\infty}))).$$

It is easy to calculate that

$$P(H_{l,l_0}) = C_{l+l_0-1}^{l-1} p_1^l (1-p_1)^{l_0};$$

$$P(M) = 1 - P(\bar{M}) = 1 - \sum_{k=0}^{l-1} C_{l+l_0}^k q_1^k \times (1-q_1)^{l+l_0-k};$$

$$P(M_k \cap M_{l-k}^{\infty}) = P(M_k) \cdot P(M_{l-k}^{\infty}) =$$

$$= C_{l+l_0}^k q_1^k (1-q_1)^{l+l_0-k} \cdot \left(\frac{q_1(1-p_1)}{p_1(1-q_1)}\right)^{l-k},$$

where the first equality in the latter expression is due to independence of M_k and M_{l-k}^{∞}, and the second one is due to the Corollary 1.

So,

$$P(F_{l,l_0}) \le C_{l+l_0-1}^{l-1} p_1^l (1-p_1)^{l_0} \times$$

$$\times \Bigg((1 - \sum_{k=0}^{l-1} C_{l+l_0}^k q_1^k \times (1-q_1)^{l+l_0-k}) +$$

$$+ \sum_{k=0}^{l-1} \left\{ C_{l+l_0}^k q_1^k (1-q_1)^{l+l_0-k} \cdot \left(\frac{q_1(1-p_1)}{p_1(1-q_1)}\right)^{l-k} \right\} \Bigg),$$

and

$$P(F(l,N)) \le \sum_{l_0=0}^{N-l} P(F_{l,l_0}) \le$$

$$\le \sum_{l_0=0}^{N-l} \Bigg[C_{l+l_0-1}^{l-1} p_1^l (1-p_1)^{l_0} \times$$

$$\times \Bigg((1 - \sum_{k=0}^{l-1} C_{l+l_0}^k q_1^k \times (1-q_1)^{l+l_0-k}) +$$

$$+ \sum_{k=0}^{l-1} \left\{ C_{l+l_0}^k q_1^k (1-q_1)^{l+l_0-k} \cdot \left(\frac{q_1(1-p_1)}{p_1(1-q_1)}\right)^{l-k} \right\} \Bigg) \Bigg],$$

the theorem is proved. □

Note that formula (17) contains binomial coefficients with large parameters l and l_0, which may take values 10^3 and more. For such values it is computationally hard to calculate the coefficients directly. But we can use the Moivre-Laplace local and integral theorem that gives a rather good approximation in our case.

So we will use the Moivre-Laplace local and integral theorem to approximate the sum.

Hence, using the Moivre-Laplace local theorem we obtain:

$$C_{l+l_0-1}^{l-1} p_1^l (1-p_1)^{l_0} \approx p_1 \cdot \frac{\varphi(\frac{l_0 p_1 + (l-1)(1-p_1)}{\sqrt{(l+l_0-1))p_1(1-p_1)}})}{\sqrt{(l+l_0-1)p_1(1-p_1)}};$$

$$C_{l+l_0}^k q_1^k (1-q_1)^{l+l_0-k} \approx \frac{\varphi(\frac{k-(l+l_0)q_1}{\sqrt{(l+l_0))q_1(1-q_1)}})}{\sqrt{(l+l_0)q_1(1-q_1)}}.$$

And using Moivre-Laplace integral theorem we obtain:

$$1 - \sum_{k=0}^{l-1} C_{l+l_0}^k q_1^k \times (1-q_1)^{l+l_0-k} =$$

$$= \sum_{k=l}^{l-l_0} C_{l+l_0}^k q_1^k \times (1-q_1)^{l+l_0-k} \approx$$

$$\approx \frac{1}{2} - \Phi\left(\frac{l-(l+l_0)q_1}{\sqrt{(l+l_0)q_1(1-q_1)}}\right) =$$

$$= \frac{1}{2} + \Phi\left(\frac{(l+l_0)q_1 - l}{\sqrt{(l+l_0)q_1(1-q_1)}}\right),$$

where $\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}$ is normal density, $\varphi(-x) = \varphi(x)$, and Φ is a Laplace function, $\Phi(x) = \int_0^x \varphi(x)dx = \int_{-\infty}^x \varphi(x)dx - \frac{1}{2}$, for $x \ge 0$, and $\Phi(-x) = -\Phi(x)$.

Using these approximations, we can provide another formulation of Theorem 1.

Theorem 2. *For the event $P(F(l,N))$, the following upper bound holds:*

$$P(F(l,N)) \le \sum_{l_0=0}^{N-l} \Bigg[p_1 \cdot \frac{\varphi(\frac{l_0 p_1 + (l-1)(1-p_1)}{\sqrt{(l+l_0-1))p_1(1-p_1)}})}{\sqrt{(l+l_0-1)p_1(1-p_1)}} \times$$

$$\times \Bigg((\frac{1}{2} + \Phi\left(\frac{(l+l_0)q_1 - l}{\sqrt{(l+l_0)q_1(1-q_1)}}\right)) +$$

$$+ \sum_{k=0}^{l-1} \Bigg\{ \frac{\varphi(\frac{k-(l+l_0)q_1}{\sqrt{(l+l_0))q_1(1-q_1)}})}{\sqrt{(l+l_0)q_1(1-q_1)}} \times$$

$$\times \left(\frac{q_1(1-p_1)}{p_1(1-q_1)}\right)^{l-k} \Bigg\} \Bigg) \Bigg]. \quad (18)$$

3. Model 2: Fork probability for an adversary with fast synchronization.

In this section we consider an advanced model for an adversary. We allow malicious miners (MMs) to be corrupted in such a way that they can be synchronized about twice as fast as the honest ones (HMs).

For some $T, k \in N$, let us define the event $M_{T,k}$ as "During exactly T TSs MMs generate exactly k blocks".

Lemma 3. *In our notations,*

$$P\left(M_{T,k}\right) = \sum_{k_2=0}^{\left[\frac{k}{2}\right]} C_T^{k_2} C_{T-k_2}^{k-2k_2} q_2^{k_2} q_1^{k-2k_2} q_0^{T-k+k_2}. \quad (19)$$

Proof. Let k_2 be the number of TSs where MMs extend their branch on two blocks.

Note that if k_2 is fixed, the event $M_{T,k}$ is just the intersection of the following events:

- MMs extend their branch by two blocks exactly in k_2 TSs;

- MMs extend their branch by one block exactly in $k-2k_2$ TSs;

- MMs generate no blocks in exactly $T-k_2-(k-2k_2)=T-k+k_2$ TSs.

The probability of such event is

$$C_T^{k_2} C_{T-k_2}^{k-2k_2} q_2^{k_2} q_1^{k-2k_2} q_0^{T-k+k_2}.$$

Then the probability of the event $M_{T,k}$ is the union of such events for all possible values of k_2 (note that any two of these events have empty intersection), and its probability is the sum of corresponding probabilities.

Finally, it is easy to see that k_2 can take values from 0 to $\left[\frac{k}{2}\right]$.

The Lemma is proved.

□

Now we are ready to formulate the main theorem about fork probability for Model 2.

Let us fix some $N \in N$ and consider the part of blockchain from TS number $t_0=1$ to TS number N. For some $l \le N$ let us define the event $F(l,N)$ as "The fork occurred that started in TS $t_0=1$ and achieved the length l before TS number N under the condition that HMs generated l confirmation blocks starting at $t_0=1$ and the fork was hidden till HMs generated these l confirmation blocks".

Theorem 3. *In our notations, the following upper estimate holds:*

$$P(F(l,N)) \le \sum_{l_0=0}^{N-l} \left[C_{l+l_0-1}^{l-1} p_1^l p_0^{l_0} \left(1 - \sum_{k=0}^{l-1} P(M_{l+l_0,k}) + \right.\right.$$

$$\left.\left. + \sum_{k=0}^{l-1} P(M_{l+l_0,k}) q^{(l-k}} \right) \right], \qquad (20)$$

where the value $q^{(l-k)}$ is defined according to (7)*, and the value $P\left(M_{l+l_0,k}\right)$ is defined according to* (19)*.*

Proof. For some $l_0 \le N-l$ let us define the event $F_{l,\,l_0}$ as "The fork with the length at least l occurred that started in TS $t_0=1$ and was hidden till HMs generated l confirmations blocks, and these blocks were generated during exactly $l+l_0$ TSs starting at $t_0=1$".

Then

$$F(l,N) \subset \bigcup_{l_0=0}^{N-l} F_{l,\,l_0} \text{ and } P(F(l,N)) \le$$

$$\le \sum_{l_0=0}^{N-l} P(F_{l,\,l_0}). \qquad (21)$$

Also let us introduce the following events:

- $H_{l,\,l_0}$ is "HMs generated l confirmation blocks during exactly $l+l_0$ TSs starting at $t_0=1$";

- $M_{l+l_0,\,\ge l}$ is "MMs generated not less than l (i.e. l or more) blocks during $l+l_0$ TSs starting at $t_0=1$";

- $M_{l+l_0,k}$ is "MMs generated exactly k $(0 \le k \le l-1)$ blocks during $l+l_0$ TSs starting at $t_0=1$";

- M_{l-k}^{∞} is "MMs ever catch up with the honest chain under the condition that in TS number $l+l_0$ they are exactly $l-k$ blocks behind".

From the definition of $F_{l,\,l_0}$, we see that

$$F_{l,\,l_0} \subset H_{l,\,l_0} \cap$$

$$\cap \left(M_{l+l_0,\,\ge l} \cup \left(\bigcup_{k=0}^{l-1} \left(M_{l+l_0,k} \cap M_{l-k}^{\infty} \right) \right) \right).$$

Next,

$$P\left(H_{l,l_0}\right) = C_{l+l_0-1}^{l-1} p_1^l p_0^{l_0},$$

$$P\left(M_{l+l_0,\,\ge l}\right) = 1 - P\left(\overline{M}_{l+l_0,\,\ge l}\right) =$$

$$= 1 - \sum_{k=0}^{l-1} P\left(M_{l+l_0,\,k}\right),$$

where $P\left(M_{l+l_0,\,k}\right)$ is defined according to (19) and

$$P\left(M_{l+l_0,k} \cap M_{l-k}^{\infty}\right) = P\left(M_{l+l_0,k}\right) P\left(M_{l-k}^{\infty}\right) =$$

$$= P\left(M_{l+l_0,k}\right) q^{(l-k)}$$

where $q^{(l-k)}$ is defined according to (7).

Then

$$P(F_{l,\,l_0}) \le C_{l+l_0-1}^{l-1} p_1^l p_0^{l_0} \left(1 - \sum_{k=0}^{l-1} P(M_{l+l_0,\,k}) + \right.$$

$$\left. + \sum_{k=0}^{l-1} P(M_{l+l_0,\,k}) \cdot q^{(l-k)} \right). \qquad (22)$$

Substituting (22) into (21), we obtain (20) and finish the proof of the theorem.

□

Note: we can also rewrite the inequality (20) in a such way:

$$P(F(l,N)) \le \sum_{l_0=0}^{N-l} \left[C_{l+l_0-1}^{l-1} p_1^l p_0^{l_0} \cdot \right.$$

$$\left. \cdot \left(1 - \sum_{k=0}^{l-1} P(M_{l+l_0,k}) \left(1 - q^{(l-k)} \right) \right) \right], \qquad (23)$$

which is easier to calculate.

And, at last, we want to simplify the condition (6).

Lemma 4. *In our notations, condition* (6) *is equivalent to the inequality*

$$(1-p)^{n\,s_H} < 2\,(1-p)^{m\,\frac{s_H}{2}} - 1.$$

Proof. In our notations,

$$P_1 = p_1 q_0;$$

$$P_{-1} = p_0 q_1 + p_1 q_2;$$

$$P_{-2} = p_0 q_2,$$

so inequality (19) can be rewritten as

$$p_0 q_1 + p_1 q_2 + 2 p_0 q_2 < p_1 q_0,$$

or

$$\frac{p_0}{1-p_0} < \frac{q_0 - q_2}{1-(q_0-q_2)},$$

or

$$p_0 < q_0 - q_2.$$

Direct calculations give us

$$q_0 - q_2 = 2\,(1-p)^{m\ \frac{s_H}{2}} - 1,$$

and, according to the definition

$$p_0 = (1-p)^{n\ s_H}.$$

The Lemma is proved. □

4. Model 3: fork probability for GHOST

In this section we assume $k = 1$, i.e.

$$p = \frac{1}{ns} \tag{24}$$

where n is the number of HMs, s is the number of attempts in one TS.

Note that in that model probability of success in one attempt (24) is 47 times larger than for two previous models.

In this section we make the following assumptions.

1) Some transaction was made at TS t_0, and there exists only one chain of blocks at this TS. Hence block B_0 with transaction was the last block of this chain. And all the next blocks generated by HMs are the "children" of block B_0, so its "weight" at some TS $t > t_0$ is equal to the number of all blocks generated by HMs from the TS t_0 till the TS t.
2) For the sake of simplicity, we assume that HMs can generate not more than 3 blocks and MMs can generate not more than 2 blocks during one TS. This restriction is not essential: the probability that HMs generate 4 or more blocks during one TS is about 0,01; the probability that MMs generate 3 or more blocks during one TS is about 0,02 in case when the ratio of MMs is about 33%.
   Without these restrictions, it seems impossible to obtain valuable results in this model.

Now we need one additional lemma.

For some $l, l_0 \in \mathbb{N}$, define the event H_{l,l_0} as "It takes exactly $l + l_0$ TSs for HMs to generate at least l blocks". In other words, H_{l,l_0} means that HMs generate not more than $l-1$ blocks during TSs $1, 2, ...l + l_0 - 1$ and generate not less than l blocks during TSs $1, 2, ...l + l_0$.

Also let us define probabilities

$$P_i = C_{sn}^i p^i (1-p)^{sn-i}, i = 0, 1, 2, 3, \tag{25}$$

where p_i is the probability that HMs generate exactly i blocks during one TS.

Lemma 5. *In our notations*

$$P(H_{l,l_0}) = P(S_{l+l_0-1} = l-1)\cdot(p_1+p_2+p_3)+ \\ + P(S_{l+l_0-1} = l-2)\cdot \\ \cdot(p_2+p_3) + P(S_{l+l_0-1} = l-3)\cdot p_3), \tag{26}$$

where

$$P(S_{l+l_0-1} = l-i) = \\ = \sum_{k_3=0}^{\left[\frac{l-i}{3}\right]} \sum_{k_2=0}^{\left[\frac{l-i-3k_3}{2}\right]} C_{l+l_0-1}^{k_3} C_{l+l_0-1-k_3}^{k_2} \times \\ \times C_{l+l_0-1-k_3-k_2}^{l-i-3k_3-2k_2} \cdot p_3^{k_3} \cdot p_2^{k_2} \cdot p_1^{l-i-3k_3-2k_2} \times \\ \times p_0^{l_0-1+i+2k_3+k_2},\ i = 1, 2, 3. \tag{27}$$

Proof. We define as $\xi_i, i \geq 1$ the number of blocks that HMs generate in TS number i. According to (25) and our assumptions,

$$\xi_i = \begin{cases} 0, & \text{with probability } p_0, \\ 1, & \text{with probability } p_1, \\ 2, & \text{with probability } p_2, \\ 3, & \text{with probability } p_3. \end{cases}$$

Also define the $S_n = \sum_{i=1}^n \xi_i$.

Now we introduce the event A_n as

$$A_n = \{min\{k \geq 1 : S_k \geq l\} = n\}.$$

In other words, A_n means that $\{S_{n-1} < l\} \cap \{S_n \geq l\}$.

In our notations, we need to find the probability $P(A_{l+l_0})$.

We define the events

$B_n^{(i)} = \{\xi_n = i\}, i = 0, 1, 2, 3$, and note that $P(B_n^{(i)}) = p_i$.

Then, according to the compound probability formula

$$P(A_n) = \sum_{i=0}^{3} P(A_n / B_n^{(i)}) P(B_n^{(i)}) = \\ = \sum_{i=1}^{3} P(A_n / B_n^{(i)}) p_i, \tag{28}$$

as $P(A_n / B_n^{(0)}) = 0$.

Next, note that for $i = 1, 2, 3$:

$$P(A_n / B_n^{(i)}) = P(l - i \leq S_{n-1} \leq l-1). \tag{29}$$

Let us find $P(S_{n-1} = l-i), i = 1, 2, 3$.

We note as k_i the number of TSs where HMs generate exactly i blocks, $i = 0, 1, 2, 3$.

Then $0 \leq k_3 \leq [\frac{l-i}{3}]$.

Note that if k_3 is fixed, then $0 \leq k_2 \leq \left[\frac{l-i-3k_3}{2}\right]$.

Next if k_3 and k_2 are fixed, then $k_1 = l-i-3k_3-2k_2$ and finally,

$k_0 = n-1-k_3-k_2-k_1 =$
$= n-1-k_3-k_2-(l-i-3k_3-2k_2) =$
$= n-1-l+i+2k_3+k_2.$

So,

$$P(S_{n-1}=l-i) = \sum_{k_3=0}^{[\frac{l-i}{3}]} \sum_{k_2=0}^{[\frac{l-i-3k_3}{2}]} C_{n-1}^{k_3} \times$$

$$\times C_{n-1-k_3}^{k_2} \cdot C_{n-1-k_3-k_2}^{l-i-3k_3-2k_2} \cdot p_3^{k_3} \cdot p_2^{k_2} \times$$

$$\times p_1^{l-i-3k_3-2k_2} \times p_0^{n-1-l+i+2k_3+k_2}. \quad (30)$$

Also, using (28) and (29), we can write that

$$\begin{aligned} P(A_n) = &\big(P(S_{n-1}=l-1)+P(S_{n-1}=l-2)+ \\ &+P(S_{n-1}=l-3)\big)\cdot p_3 + (P(S_{n-1}=l-2)+ \\ &+P(S_{n-1}=l-1))\cdot p_2 + P(S_{n-1}=l-1)p_1 = \\ &= P(S_{n-1}=l-1)(p_1+p_2+p_3)+ \\ &+P(S_{n-1}=l-2)(p_2+p_3)+ \\ &+P(S_{n-1}=l-3)p_3, \end{aligned} \quad (31)$$

and formulas (30) and (31) finish the proof of the lemma, when $n = l + l_0$. □

To formulate the main result, we also need formula (19) from Lemma 3, but for values q_0, q_1, q_2 defined for Model 3 in (2).

Theorem 4. *Let the event $F(l,N)$ be the same as defined in Models 1 or 2. Then*

$$P(F(l,N)) \leq$$

$$\leq \sum_{l_0=0}^{N-l} \big[P(H_{l,l_0}) \times (1 - \sum_{k=0}^{l-1} \{P(M_{l+l_0,k}) \cdot (1-Q^{(l-k)})\})\big],$$

where $P(M_{l+l_0,k})$ is as defined in (19) *and $P(H_{l,l_0})$ is as defined in* (26) *using values* (2) *and* (3).

The proof of this theorem is just the same as the proof of Theorem 3, but the probabilities of events H_{l,l_0} and $M_{l+l_0,k}$ take other values that in (20).

5. Comparison of confirmation blocks' numbers for different methods

The Table 1 shows the number z of block confirmations for attack success probability of 0.001 for various values of the adversarial hashrate q, determined by the methods developed by S.Nakamoto [1], M.Rosenfeld [2], C. Grunspan and R.Perez-Marco [3], compared to our results obtained for Bitcoin consensus in the network with equal delays both for honest miners and attacker nodes (Model 1), for Bitcoin consensus on the network with faster (2x) adversarial synchronization (Model 2) and for the GHOST protocol (Model 3).

For this computation, we took $s_H = 1000$ and $s_M = s_H$ for Model 1 and Model 3; for Model 2, we took $s_M = \frac{s_H}{2}$ that means twice as fast synchronization for adversarial nodes; $n = 1000$ and $N = 17000$ (these parameters provide sufficiently good accuracy due to attack success probability value saturation; further increasing of N, shows no changes in block confirmations number given in the table). We took the ratio of block generation time to network block propagation time as $k = 47.6$ for Bitcoin, Model 1 and Model 2, and $k = 1$ for GHOST, Model 3 [10].

To verify theoretical results independently, we also performed direct simulation of attacks in the software and obtained results that are very close to the ones given in the table.

Though our method for Model 1 is quite different from the methods proposed by M.Rosendeld and C.Grunspan, we got exactly the same numbers for block confirmation number. Full coincidence of results provides additional evidence of right approach taken in the papers.

For the Model 2, we can see that even 2x faster adversarial synchronization gives an advantage for an attacker only for high adversarial hash rate (0.35+).

The GHOST rule requires the number of confirmation blocks comparable to Bitcoin. Taking into account much shorter time between blocks for GHOST, that gives advantage to this consensus protocol by providing the same level of blockchain security in shorter time.

Conclusions

The number of transaction confirmation blocks is important for application properties of a cryptocurrency and is closely related to average time of receiving and accepting of payments. The shortest confirmation time for the same level of transaction security provides the best user properties for cryptocurrency.

Papers that provide a number of transaction confirmation blocks for Bitcoin use models with implicit assumption of prompt spreading of Bitcoin blocks over the network that leads to conditions that are not always the case for the real world conditions of cryptocurrencies application. Papers that take into account delays of message delivery on peer-to-peer networks, provide proofs of asymptotic estimates on reaching of main blockchain properties, with no specific values of numbers of transaction confirmation blocks.

We developed three methods for determination of the required number of confirmation blocks for Bitcoin and GHOST. The first method uses a model that considers equal network delays for message delivery on Bitcoin peer-to-peer network both for honest and malicious miners. The second one is for Bitcoin and assumes that an attacker may have faster synchronization for attack optimization. The third method allows to determine required number of confirmation blocks for the GHOST protocol. It is the first strict theoretical method (to our knowledge) that allows obtaining of these values for the GHOST.

Compared to other existing methods, in the conditions of equal delays of synchronization for honest miners and adversarial nodes, our method gives the same numbers as the known results by M.Rosenfeld and C.Grunspan, et.al, though uses quite different approach

Table 1. The number z of block confirmations for attack success probability of 0.001 for various values of the adversarial hashrate q for different models

q	S.Nakamoto	M.Rosenfield	C.Grunspan and R.Perez-Marco	Model 1 (Bitcoin)	Model 2 (Bitcoin, fast adv. synch.)	Model 3 (GHOST)
0.1	5	6	6	6	6	6
0.15	8	9	9	9	9	8
0.2	11	13	13	13	13	12
0.25	15	20	20	20	20	18
0.3	24	32	32	32	32	28
0.35	41	58	58	58	59	49
0.4	81	133	133	133	136	101

(also taking into account message delivery delays). In the model with 2x faster adversarial synchronization, an attacker may gain an advantage only controlling high hash rate (0.35+).

According to our method, the GHOST protocol requires the number of confirmation blocks, comparable to Bitcoin. But having much shorter time between blocks, GHOST has advantage by providing the same level of blockchain security in shorter time.

References

[1] S. Nakomoto, "A peer-to-peer electronic cash system," *online*, 2008.

[2] M. Rosenfeld, "Analysis of hashrate-based double-spending," *arXiv preprint*, 2014.

[3] C. Grunspan and R. Pérez-Marco, "Double spend races," *CoRR*, vol. abs/1702.02867, 2017.

[4] C. Pinzon and C. Rocha, "Double-spend attack models with time advantange for bitcoin," *Electronic Notes in Theoretical Computer Science*, vol. 329, pp. 79–103, 2016.

[5] J. A. Garay, A. Kiayias, and N. Leonardos, "The bitcoin backbone protocol: Analysis and applications," *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applicaitons of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part II,*, pp. 281–310, 2015.

[6] R. Pass, L. Seeman, and A. Shelat, "Analysis of the blockchain protocol in asynchronous networks," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 643–673, Springer, 2017.

[7] J. A. Garay, A. Kiayias, and N. Leonardos, "The bitcoin backbone protocol with chains of variable difficulty.," *IACR Cryptology ePrint Archive*, vol. 2016, p. 1048, 2016.

[8] Y. Sompolinsky and A. Zohar, "Secure high-rate transaction processing in bitcoin," *Financial Cryptography and Data Security - 19th International Conference, FC 2015, San Juan, Puerto Rico, January 26-30, 2015, Revised Selected Papers*, 2015.

[9] Y. Sompolinsky and A. Zohar, "Accelerating bitcoin's transaction processing. fast money grows on trees, not chains," *IACR Cryptology ePrint Archive*, vol. 2013, p. 881, 2013.

[10] A. Kiayias and G. Panagiotakos, "Speed-security tradeoffs in blockchain protocols," *Cryptology ePrint Archive, Report 2015/1019*, 2015.