Number of confirmation blocks for Bitcoin and GHOST consensus protocols on networks with delayed message delivery

Lyudmila Kovalchuk[1,2]

Joint work with Dmytro Kaidalov[1], Andrii Nastenko[1],

Olexiy Shevtsov[1], Mariia Rodinko[1,3], Roman Oliynykov[1,3]

{lyudmila.kovalchuk, dmytro.kaidalov, andrii.nastenko, oleksiy.shevtsov, mariia.rodinko, roman.oliynykov}@iohk.io

[1] Input Output HK, Hong Kong

[2] National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Kyiv, Ukraine

[3] V.N. Karazin Kharkov National University, Kharkiv, Ukraine

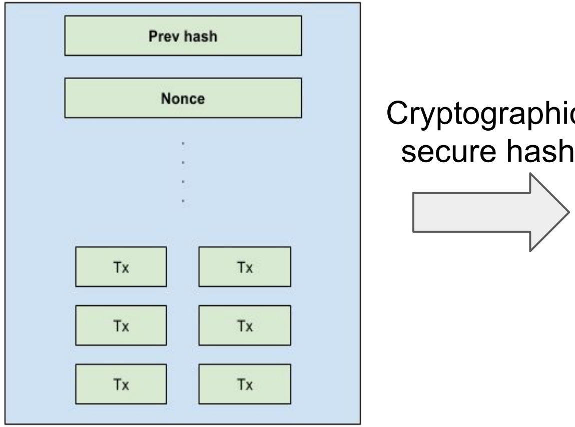
June 15th, 2018

Proof-of-Work consensus algorithm

In PoW blockchain systems an ability to add next block is provided to the node that generated a block with a hash of data that is below some **target,** which requires many attempts (computational work).

As far as all data in a block is valid, all network participants will consider an entire block as valid and add it to their local blockchains.

Block



Cryptographically secure hashing

0x0000000000008e962c6a410cfa73a829d59e569f8203a0cfe... **< target**

target = 0x000000000000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF...

Security: to attack the network, the adversary must do bigger amount of work than honest nodes (that is very costly and makes the attack economically senseless) or be able to break the cryptographic hash (SHA-256)

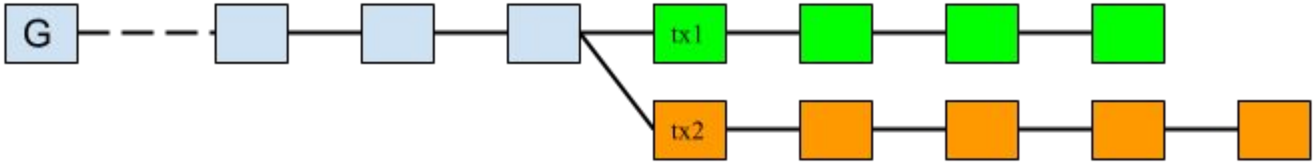
Proof-of-Work consensus algorithms

The most widely spread PoW systems:

- **Bitcoin**;
- Litecoin;
- Ethereum;
- ZCash
- Dash;
- etc.

Double-spend attack

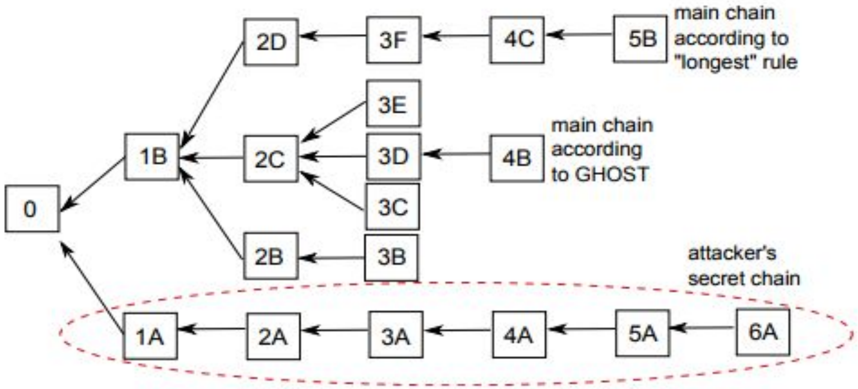
As it follows from the name, the whole idea of a double-spend attack is to spent the same coins twice. In general, it implies that someone pays for some goods, but after receiving them, makes the cryptocurrency network to revert the payment so both goods and coins are in the hands of an attacker.



The Greedy Heaviest-Observed Sub-Tree (GHOST)

The big problem of Bitcoin: scaling in order to support the higher volume of transactions

The solution: to decrease a block generation time keeping the same security level due to a new rule for the selection of the main chain in the block tree: blocks that are off the main chain can still contribute to its weight (figure below[1]).


2D
3F
4C
5B
main chain according to "longest" rule
3E
1B
2C
3D
4B
main chain according to GHOST
0
3C
2B
3B
attacker's secret chain
1A
2A
3A
4A
5A
6A

[1] Yonatan Sompolinsky and Aviv Zohar. Secure High-Rate Transaction Processing in Bitcoin

Analysis of Bitcoin Double-Spend Attack

There are several well-known mathematical models that analysis the possibility of a double spend attack in Bitcoin:

- The model of S. Nakomoto

$$1-\sum_{k=0}^{z}\frac{\lambda^{k}e^{-\lambda}}{k!}\left(1-(q/p)^{(z-k)}\right) \qquad \lambda=z\frac{q}{p}$$

- The model of M. Rosenfeld

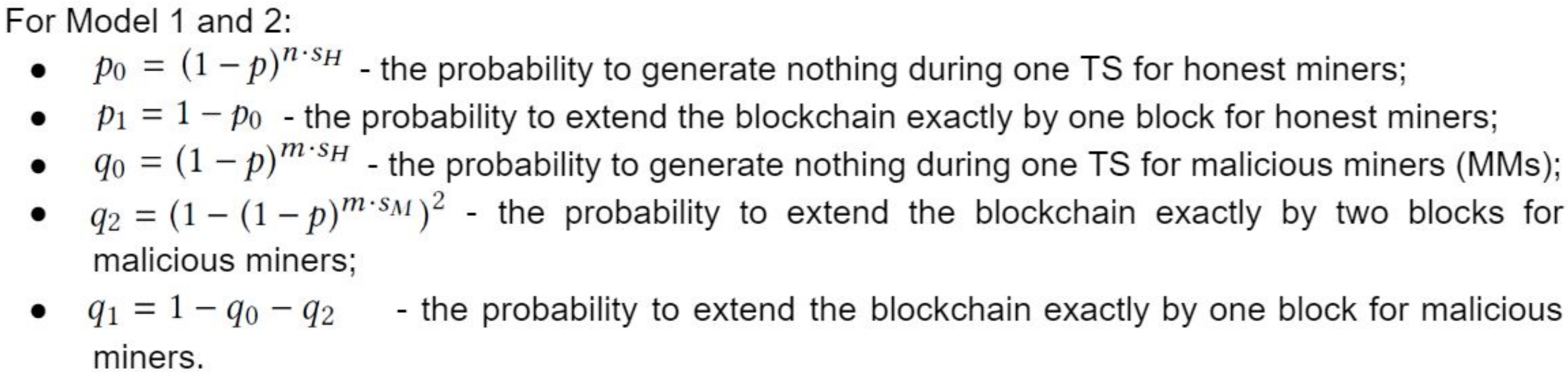
$$\begin{aligned} r &= \sum_{m=0}^{\infty} P(m)a_{n-m-1} \\ &= \sum_{m=0}^{n-1}\binom{m+n-1}{m}p^{n}q^{m}\left(\min(q/p,1)\right)^{n-m}+\sum_{m=n}^{\infty}\binom{m+n-1}{m}p^{n}q^{m} \\ &= \begin{cases} 1-\sum_{m=0}^{n}\binom{m+n-1}{m}(p^{n}q^{m}-p^{m}q^{n}) & \text{if } q<p \\ 1 & \text{if } q\geq p \end{cases} \end{aligned}$$

- Others (the model of C. Grunspan, the model of C. Pinzon et al.)

Preliminary notations (I)

- Timeslot (TS) - the period of synchronization, i.e. the amount of time needed to share a block between independent miners;
- t_1 - the period of network synchronization for honest miners (HMs);
- t_2 - the time needed for one attempt of block generation;
- s_H - the ratio t_1 / t_2;
- $s_M = s_H$ (for the first model);
- $s_M = s_H / 2$ (for the second one);
- $s_M = s_H = s$ (for the third model);
- k - the ratio of block generation time to network block propagation time;
- p - the probability to generate a block by one miner in one attempt (we assume $p = 1 / k{\cdot}n{\cdot}s_H$);
- n - the number of honest miners;
- m - the number of malicious miners (we assume that $m < n$, so honest miners have majority).

Preliminary notations (II)



For Model 1 and 2:

- $p_0 = (1-p)^{n\cdot s_H}$ - the probability to generate nothing during one TS for honest miners;
- $p_1 = 1 - p_0$ - the probability to extend the blockchain exactly by one block for honest miners;
- $q_0 = (1-p)^{m\cdot s_H}$ - the probability to generate nothing during one TS for malicious miners (MMs);
- $q_2 = (1-(1-p)^{m\cdot s_M})^2$ - the probability to extend the blockchain exactly by two blocks for malicious miners;
- $q_1 = 1 - q_0 - q_2$ - the probability to extend the blockchain exactly by one block for malicious miners.

For Model 3:

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

$$q_i = \binom{ms}{i} p^i (1-p)^{ms-i} \; for \; i = 0, 1;$$

$$q_2 = 1 - q_0 - q_1$$

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

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).

Model 1. Fork probability for an adversary with ordinary synchronization

Let’s 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 }.

For the event $F(l,N)$ the following upper bound holds:

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

$$\leq \sum_{l_0=0}^{N-l}\left[\binom{l+l_0-1}{l-1}p_1^l(1-p_1)^{l_0}\cdot\right.$$

$$\cdot\left((1-\sum_{k=0}^{l-1}\binom{l+l_0}{k}q_1^k\times(1-q_1)^{l+l_0-k})+\right.$$

$$+\sum_{k=0}^{l-1}\left\{\binom{l+l_0}{k}q_1^k(1-q_1)^{l+l_0-k}\cdot\right.$$

$$\left.\left.\left.\cdot\left(\frac{q_1(1-p_1)}{p_1(1-q_1)}\right)^{l-k}\right\}\right)\right].$$

After approximation:

$$P(F(l,N)) \leq \sum_{l_0=0}^{N-l}\left[p_1\cdot\frac{\varphi\left(\frac{l_0p_1+(l-1)(1-p_1)}{\sqrt{(l+l_0-1))p_1(1-p_1)}}\right)}{\sqrt{(l+l_0-1)p_1(1-p_1)}}\times\right.$$

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

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

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

$\varphi(x)$ is a normal density, $\varphi(-x) = \varphi(x)$;

Φ is a Laplace function.

Model 2. Fork probability for an adversary with fast synchronization (I)

For some T, $k \in N$, let’s define the event $M_{T,k}$ as “During exactly T timeslots malicious miners generate exactly k blocks”.

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

Let’s 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 honest miners generated l confirmation blocks starting at t_0 = 1 and the fork was hidden till honest miners generated these l confirmation blocks”. In our notations, the following upper estimate holds:

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

where the value $q^{(l-k)}$ is defined according to the expressions below.

Model 2. Fork probability for an adversary with fast synchronization (II)

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

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

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

and define RVs $\{\zeta_i, i \geq 1\}$, as $\zeta_i = \xi_i - \eta_i$.

The probability distribution of ζ_i, $i \geq 1$ is

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

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

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

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

If the condition $P_{-1} + 2P_{-2} < P_1$ holds, then

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

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

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

Model 3. Fork probability for GHOST

Assumptions:

- k = 1, i.e. p = 1 / ns.
- 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. 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.
- HMs can generate not more than 3 blocks and MMs can generate not more than 2 blocks during one TS. This restriction is not very 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%.

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

$$P(F(l,N)) \leq \sum_{l_0=0}^{N-l} \Big[P(H_{l,l_0})\times$$

$$\times(1-\sum_{k=0}^{l-1}\{P(M_{l+l_0,k})\cdot(1-q^{(l-k)})\})\Big]$$

$P(M_{l+l0,k})$ is as defined for Model 2;

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

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

Comparison of confirmation blocks’ numbers for different methods (I)

For the computation, we took:

- s_H = 1000 and $s_M = s_H$ for Model 1 and Model 3;
- $s_M = s_H$ / 2 for Model 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, etc. shows no changes in block confirmations number given in the table);
- k = 47.6 - the ratio of block generation time to network block propagation time as for Bitcoin, Model 1 and Model 2;
- k = 1 for GHOST, Model 3.

Comparison of confirmation blocks' numbers for different methods (II)

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

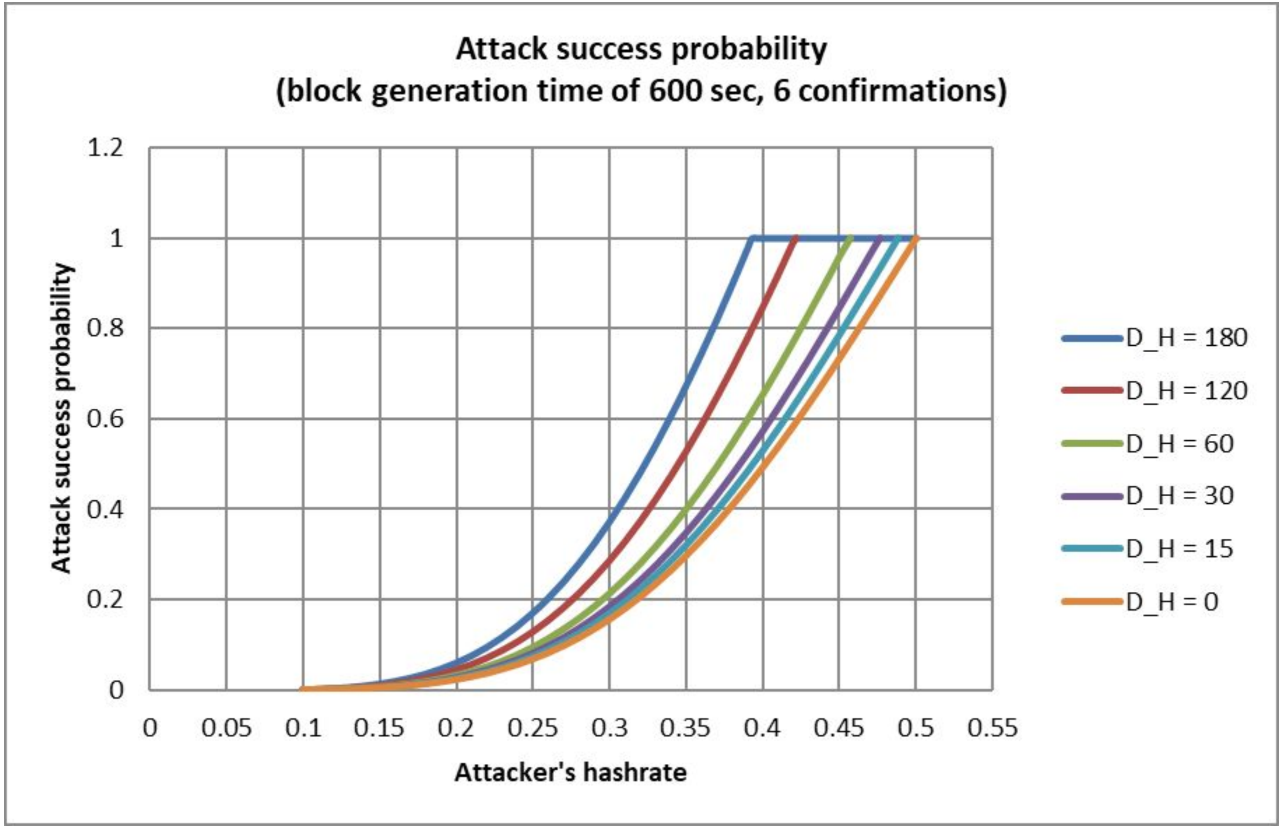
q	S.Nakamoto	M.Rosenfield	C.Grunspan	Model 1 (Bitcoin)	Model 2 (fast adv.)	Model 3 (GHOST)
0.1	5	6	6	6	6	6
0.15	8	9	9	9	9	8
0.2	11	13	13	13	13	12
0.25	15	20	20	20	20	18
0.3	24	32	32	32	32	28
0.35	41	58	58	58	59	49
0.4	81	133	133	133	136	101

Comparison of confirmation blocks' numbers for different synchronization time

Table 2. The results for block generation time of 600 sec and different values of malicious hashrate and synchronization time

q	$D_H = 0$	$D_H = 5$	$D_H = 15$	$D_H = 30$	$D_H = 60$
0.1	6	6	7	8	10
0.15	9	9	11	13	19
0.2	13	14	17	22	42
0.25	20	22	28	43	172
0.3	32	37	54	113	$P_{success} = 1$
0.35	58	74	137	$P_{success} = 1$	

Attack success probability for different synchronization time



Conclusions (I)

- We developed three methods for determination of the required number of confirmation blocks for Bitcoin and GHOST that took into account the real world conditions of peer-to-peer network synchronization of cryptocurrencies. 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.

Conclusions (II)

- 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 (also taking into account message delivery delays). In the model with 2x faster adversarial synchronization, an attacker may gain an advantage having less than a half of hashing power (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.

- If an adversary is highly-synchronized, a double-spend attack may have a success with probability 1, even if the ratio of adversary is much less than 50%.