Difficulty Control for Blockchain Systems

Dmitry Meshkov[a], Alexander Chepurnoy[a]

[a] *IOHK Research*

Abstract

TODO

Keywords: Blockchain, Decentralized consensus, Peer-to-peer networks, Proof-of-Work

1. Introduction

Blockchain systems have attracted a growing amount of interest from various communities after publication of Bitcoin whitepaper [1] in 2008. Bitcoin security relies on the distributed protocol that maintains the blockchain, called mining, in which network nodes tries to solve computational puzzle. Other blockchain systems may relies on different computational puzzles [??] or even on virtual mining [??], while all of them use some algorithm that changes puzzle difficulty dynamically. This algorithm for retargeting the difficulty is required to make blockchain system predictable and fix latency between blocks.

Fixed latency between blocks is important for several reasons. Too often blocks leads to situation, when for a lot of miners block propagation time become bigger, then latency between blocks. This leads to significant increasing of number of blockchain forks that complicates the consensus[2] and reduce effective hash rate in blockchain system. On the other hand, increasing of the latency between blocks leads to decreasing of the network throughput[3] and may be critical for high-loaded blockchain systems like bitcoin, where blocks are already 70% full today[4]. Increasing latency by 50Moreover this will lead

Email address: `dmitry.meshkov@iohk.io` (Dmitry Meshkov)

to infinite growth of unconfirmed transactions pool, meaning it is likely that most bitcoin transactions will not even relay, much less confirm.

Most of blockchain systems relies on quite a naive difficulty retargeting algorithm, that assumes, that total computational power, involved in mining process, don't change over time. Using more complicated retargeting algorithms with incorrect assumptions[5] may lead to incorrect time interval between blocks even for simple case of constant hash rate as, for example, observed in Nxt, where observed mean time between blocks is 2 times bigger, then expected in whitepaper[6]. Moreover, too ofter difficulty recalculation leads to wide distribution of time intervals between blocks and makes blockchain system unpredictable[5]. Varying network computational power makes this algorithms inefficient for difficulty recalculation, e.g. continuous growth of computational power leads to decreasing mean latency between blocks and average block time in Bitcoin network is 1.07 times lower, then expected. Noteworthy, that exponential growth of computational power, which is the situation observed in practice in accordance with Moores law[7], is the absolutely worst case (regarding the maximal block rate) possible for Bitcoins difficulty retargeting algorithm[8].

Original Bitcoin white-paper, states that the security of the system is guaranteed as long as there is no attacker in possession of half or more of the total computational power used to maintain the system Nakamoto2008. However difficulty is not constant, and can be manipulated by the attacker. The Difficulty Raising Attack, introduced in [9], enables the attacker to discard n-depth block, for any n and any attacker hash power, with probability 1 if he is willing to wait enough time. The fact that there is no way to determine whether a block have been computed on its declared time or not, have been used as part of other attacks [10, 11].

Thus difficulty recalculation is small, but important part of blockchain systems **TODO**

2. Bitcoin Mining

3. The Difficulty Control Attack

4. An Improved Difficulty Control

5. Simulations

6. Conclusions

References

[1] S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System (2008). arXiv:43543534534v343453, doi:10.1007/s10838-008-9062-0. URL http://s.kwma.kr/pdf/Bitcoin/bitcoin.pdf

[2] C. Decker, R. Wattenhofer, Information propagation in the bitcoin network, in: Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on, IEEE, 2013, pp. 1–10.

[3] R. Böhme, T. Okamoto, On scaling decentralized blockchains, 2016. URL http://fc16.ifca.ai/bitcoin/papers/CDE+16.pdf

[4] B. Armstrong, What happened at the satoshi roundtable (2016). URL https://medium.com/@barmstrong/what-happened-at-the-satoshi-roundtable-6c11a10d8cdf#.pvoqt832u

[5] andruiman, Nxt forging algorithm: simulating approach. URL https://www.scribd.com/doc/243341106/Nxt-forging-algorithm-simulating-approach

[6] andruiman, Nxt forging algorithm: simulating approach. URL http://wiki.nxtcrypto.org/wiki/Whitepaper:Nxt

[7] G. E. Moore, Cramming more components onto integrated circuits, reprinted from electronics, volume 38, number 8, april 19, 1965, pp. 114 ff., IEEE Solid-State Circuits Newsletter 3 (20) (2006) 33–35.

[8] D. Kraft, Difficulty control for blockchain-based consensus systems, Peer-to-Peer Networking and Applications (2015) 1–17.

[9] L. Bahack, Theoretical bitcoin attacks with less than half of the computational power, arXiv preprint arXiv:1312.7013.

[10] The timejacking attack.
URL `http://culubas.blogspot.com`

[11] ArtForz, The time wrapping attack.
URL `https://bitcointalk.org/index.php?topic=43692.msg521772#msg521772`