Discrete event simulation of different selfish mining strategies in Bitcoin

Master Thesis Proposal

Advisor: Edgar Weippl

June 22, 2017

1 Problem statement

The cryptocurrency Bitcoin started back in the year 2008 with the release of the Bitcoin white paper [6] and reached as of today a market capitalization of over 20 billion dollars [2, 1]. Internally the Bitcoin cryptocurrency records all transactions in a public ledger called the blockchain. The blockchain is basically an immutable linked list of blocks where a block contains multiple transactions of the cryptocurrency. In Bitcoin, each block needs to contain a so-called proof of work (PoW) which is the solution to a costly and time-consuming cryptographic puzzle. Miners connected in a peer-to-peer network compete with their computation power to find solutions to the puzzle and hence to find the next block for the blockchain. Finding a block allows the miners to add a transaction to the block and gives them a certain amount of bitcoins out of thin air. After a block is found by a miner all other miners should adopt to this new tip of the chain and try to find a new block on top.

Eyal and Sirer showed that miners have an incentive to not follow the protocol as described depending on their connectivity in the peer-to-peer network and computation power. By implementing a so-called selfish mining strategy a miner can obtain more revenue than his actual proportion of computational power in the network. In general, the miner simply does not shares found blocks with the others and secretly mines on his own chain. At some point, the public miners may catch up with their public chain. In this case, the selfish miner publishes his blocks. If his chain is longer then the public chain, he is able to overwrite all blocks found by the honest miners. If the two chains have the same length the miners are split into two parts where one part is mining on the public tip and the other part is mining on the now public-private tip. In general, the selfish miner achieves that the other miners are wasting their computational power on blocks which will not end up in the longest chain. This raises his shares of computational power and therefore also his revenue.

Further research [7, 8] explored different modifications of the original selfish mining algorithm by Eyal and Sirer and found slightly modifications of the algorithm which are more profitable under certain circumstances. For example, it could make sense for the selfish miner to even trail behind the public chain.

To proof the profitability of selfish mining different approaches were applied. Eyal and Sirer proved the concept theoretically with state machines and gave results of a closed-source simulation. Other researchers mainly used Markov Chains especially Markov Decision Processes to simulate selfish mining [7, 8, 5]. Despite all this research there are no publications using discrete event simulation (DES) [4] to simulate selfish mining. This simulation type has the advantage to simulate the real components of Bitcoin and is not introducing any abstraction. Therefore the results are more accurate and the simulation may give new insights in the area of selfish mining.

2 Expected result

The expected outcome of this thesis is a DES simulation of different selfish mining strategies. The selfish mining strategies used in the simulation are:

- selfish mining [3]
- lead stubborn mining [7]
- trail stubborn mining [7]
- equal-fork stubborn mining [7]

These strategies are combined with different network topologies and block discovery series. The result of the simulations should show which strategy is the best strategy for a certain combination of network topology and block discovery series.

Furthermore, the simulations should emphasise the recent work in the area of selfish mining and show that the current implementation of Bitcoin protocol is vulnerable against different selfish mining strategies.

3 Methodology and approach

First, the different mining strategies selfish, lead stubborn, trail stubborn and equal-fork stubborn mining from Nayak et al. and Eyal and Sirer need to be implemented. This is achieved by implementing a proxy which eclipses a normal Bitcoin client from the other nodes in the network. Now, if a block is found the proxy decides, depending on his selfish mining strategy, if a block should be transmitted from the eclipsed node to the rest of the network or vice versa. The design pattern proxy makes it possible to implement the selfish mining strategies without altering the reference implementation of Bitcoin (bitcoind) and is therefore preferred over an implementation directly in bitcoind.

In the next step, a DES simulation program is implemented. To be able to control when a certain node finds a block all Bitcoin nodes are executed in test mode. In test mode the real proof of work algorithm is disabled and every node accepts a command which lets the node create immediately a new block. With this functionality, it is possible to define a block discovery series which basically reflects the computation power of each node. The more blocks are found by a node the more simulated computation power the node has. Additionally to the block generation, the simulation program should also control the network topology and hence the connectivity of each node. For the simulation run, it is important that the connectivity of the nodes stays the same to make the results better comparable. This should be achieved by setting the connections from the nodes by the simulation program itself which is in contrast to normal behaviour. Normally Bitcoin nodes share their connections with other nodes over the Bitcoin protocol and try to improve the connectivity over time.

After the implementation of the selfish mining strategies and the DES simulation program, the mining strategies are simulated. Different settings for the connectivity and distribution of computation power are used to compare the profitability of the selfish mining strategies to the normal, honest mining.

4 State-of-the-art

bitcoin

selfish mining each paper countermeasures no countermeasures implemented

5 Relation to "Software Engineering and Internet Computing" curriculum

Bitcoin is a relatively new topic and hence there are no concrete subjects teaching this technology in the current curriculum. But under the hood Bitcoin technically is just a composition of many areas in computer science part of the curriculum. One area is cryptography which is needed to secure the Bitcoin system. Cryptography enables the proof of work algorithm in the mining process by providing hash functions. Furthermore, cryptographic signatures are used to secure the bitcoins held by the different users of the system. Another area which is part of the curriculum is computer networks. All nodes part of the Bitcoin network are connected over a large peer-to-peer network which enables them to exchange messages of the Bitcoin protocol.

References

[1] Bitcoin (btc) market capitalization - Coindesk. http://www.coindesk.com/data/bitcoin-market-capitalization/. Accessed: 2017-06-21.

[2] Bitcoin (btc) market capitalization - CryptoCurrency Market Capitalizations. https://coinmarketcap.com/currencies/bitcoin/. Accessed: 2017-06-21.

[3] I. Eyal and E. G. Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *International Conference on Financial Cryptography and Data Security*, pages 436–454. Springer, 2014.

[4] G. S. Fishman. Principles of discrete event simulation.[book review]. 1978.

[5] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun. On the security and performance of proof of work blockchains. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 3–16. ACM, 2016.

[6] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[7] K. Nayak, S. Kumar, A. Miller, and E. Shi. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, pages 305–320. IEEE, 2016.

[8] A. Sapirshtein, Y. Sompolinsky, and A. Zohar. Optimal selfish mining strategies in bitcoin. In *International Conference on Financial Cryptography and Data Security*, pages 515–532. Springer, 2016.