Block Propagation Applied to Nakamoto Networks

Dragan Boscovic, Nakul Chawla, Darren Tapp

June 28, 2018

Abstract

Not long after the introduction of Nakamoto networks such as Bitcoin [Nak08], there has been much debate about the possibility and ability of scaling these networks. We estimate the throughput of these networks under a variety of different configurations and implementations. Of particular emphasis, we study how different block propagation techniques effect this throughput limit.

1 Introduction

We define a Nakamoto network to be a network of actors, which can be computer nodes, that agree on information and have a degree of certainty that other nodes agree on this information. We explore the Nakamoto network supporting the digital cash application, Dash.

Our intent is to answer two questions:

1. Can Dash scale, and how?
2. What is a practical limit on Dash scaling.

To answer these questions we simulated the Dash network running in different use cases. Since bandwidth is considered to be the first limiting factor, we focused on the different bandwidth requirements of different block propagation techniques. Specifically we simulated the Dash network employing three block propagation protocols namely a) traditional block propagation where the block is broadcast in full, b) compact [Cor16] block propagation and c) Xtreme thinblocks (xthin) [Tsc16] block propagation.

We found that xthin block propagation can support the Dash network with a capacity at least an order of magnitude larger than the original Dash protocol. We expect that compact blocks can also support this capacity. However, compact blocks seem to not work as well at larger block sizes. We also found that traditional block propagation ran into two limits at scale. One limit involving economic considerations of miners and another limit where the network became unstable.

2 Simulations

2.1 Methodology

Our Bitcoin simulator is an NS3 based simulator that was originally built for the research shown in [GKW$^+$16]. The simulator demonstrates the workings of Bitcoin, Dogecoin and Litecoin, but bases its research on each block being the fundamental/rudimentary component.

We found that simulations that merely went down to the block level were not detailed enough for our purpose. In order to reliably model a consensus network with block propagation speedups such as compact or xthin we must understand these networks down at the level of transactions. We found it necessary to completely rework these simulations for our purpose. The resulting simulator code is available at [Cha18]. With the addition of newer block propagation methods, we can now simulate bigger block sizes than the original simulator can.

Our simulations assumed 8 major mining pools. These pools were connected to each other.

This simulator allows us to divide the world up into different zones. Internet speeds can then be assigned to each region. We consider our internet bandwidth estimates to be conservative and consistent with what is available with a general VPS service. The simulations we ran were applied to networks with at least 6000 nodes. The number 6000 is considered to be a good upper estimate for the number of masternodes, currently, and for the next few years. General architecture of nodes is also consistent with dual core machines with 2GB of RAM. To account for variance, simulations were run long enough to simulate at least 700 blocks.

Further modifications of the simulator include new modes simulating a network with compact or xthin block propagation. Our xthin implementation used bloom filters with the murmur hash. The ability to flag a node as a masternode was added.

For simplicity, the orphan rate was calculated by taking all orphan blocks in our simulation and dividing by the total number of blocks. When making economic calculations, an orphan rate of one half of this is needed to account for the possibility of an orphan block actually being stale, that is mined on a previous blocks after a new block has been found and broadcast.

2.2 Traditional Block Propagation

Our simulations measured the orphan rate and the median block propagation time. The median block propagation time is the median time it takes to propagate over the whole network. The longest orphan chain is the longest chain of blocks that are all orphaned. Our simulations produced the following results:

Blocksize	Orphan Rate	Median Block Propagation Time	Longest Orphan Chain
100 kB	0.362%	2.02 seconds	1
200 kB	0.186%	2.95 seconds	1
750 kB	2.52%	8.51 seconds	1
1 MB	2.27%	11.01 seconds	1
1.5 MB	5.68%	17.80 seconds	2
2 MB	7.53%	18.92 seconds	2
4 MB	16.6%	70.20 seconds	3
10 MB	91.4%	18,879.4 seconds	47

From these results we can see that block propagation times increase as the block size increases. For 10MB the block propagation time is very large (over 1,700× the median time of 1MB block, and over 100× longer than the ∼150 second target). The longest orphan chain of 47 blocks suggests that the network failed to come to consensus. Given that over 90% of the blocks are orphaned, it is reasonable to conclude that the simulated network had nodes working on other chains. The simulated network with 4MB blocks most likely had a clear consensus as the longest orphan chain was three blocks. The simulation with 100kB blocks is believed to overestimate the orphan rate as the network was running over capacity and a large mempool actually hurt the performance of each node. Throwing out the 100kB simulation and the 10MB simulation as outliers. Linear regression approximates the orphan rate to increase by 4.3% per MB. This regression has a coefficient of determination of .993 which suggests a linear model works well in the limited range from 0MB to 4MB. For economic considerations, we approximate what we call the economic orphan rate to be half of this 4.3% or 2.15%. This reduction in orphan rate helps account for the fact that some orphan blocks will be stale. A stale block is an orphan block was mined after a competing block was already mined at the same height. This reduction is to account for the fact that blocks could still be stale even if a particular miner mines a smaller blocks.

We approximate how economic factors will also effect the capacity of the Dash network. As the network scales, the economic incentive of miners to include more transactions in a block should be considered. Assuming all transactions have a .01 DASH per MB fee density, a mining reward of 1.67 DASH and an economic orphan rate increase of 2.15% per MB. These results suggest that blocks over 896 kB would be uneconomical to mine. That is, the transaction fee does not compensate miners for the increased orphan rate.

We call this limit the economic limit. A limitation on block size that restricts the upper bound based on the assumption that miners behave rationally and are in search of profit. So that even with no limit on block size in code miners would not mine blocks over this limit. Miners that stay under the economic limit would be at an advantage from a economic perspective, and we would not expect the network fragmentation which our simulations show at 10MB. Once the orphan rate becomes too large miners that mine smaller blocks will be able to secure their reward with more certainty.

We expect that different block propagation techniques will allow for larger blocks. However, with traditional block propagation, we would recommend that the coded limit on block size not be above 5MB and not expect a capacity throughput much over 890kB per block. Note that users that include a higher fee might make blocks over 890kB economic to mine.

These results are helpful for us to identify when the network is not performing as expected. Our simulations assumed that all miners mined blocks of the same size. We expect that in a environment where blocks are propagating slowly and no clear consensus is obtained that miners would adjust to mine smaller blocks. Economic incentives are such that miners would adjust block size to allow a clear consensus.

2.3 Compact and Xthin Block Propagation

We chose block sizes of 750kb, 1MB, 1.5MB, 2MB, 4MB, and 10MB. We adapted the simulator to allow for simulations of compact block and xthin block propagation. Of note, simulation of compact blocks had rather large RAM requirements. We were not able to simulate the 10MB chain for compact blocks. Simulations with 20 blocks allow us to estimate the orphan rate to be between 0% and 23.1%. No network fragmentation was observed, or is expected. These simulations required adjusting the simulator to go down to the transaction level. This explains the increased ram requirements.

Orphan rates found by our simulator are summed up as follows.

Blocksize	Compact Orphan Rate	Xthin Orphan Rate
750 kB	0%	0%
1 MB	0%	0%
1.5 MB	0%	0%
2 MB	0%	0.14%
4 MB	.80%	0.56%
10 MB	N/A	0.93%

It is clear that xthin will allow for 10MB blocks without a substantial orphan rate. We also note a 20 fold reduction in the orphan rate for the 4MB chain when compact blocks are used. Almost a 30 fold reduction for xthin.

The compact block protocol has several steps. After a compact block is sent it's possible that the receiving node may have to request transactions not known by that node. On the Bitcoin network it is estimated that this request is needed 0.6% of the time. We expected that this would increase as blocksize increases. Investigation of our simulations found this to be the case.

Even though the compact block simulation did not complete we expect that the Dash network can support 10MB blocks with compact block propagation. It is clear that the Dash network can be healthy with 10MB blocks using xthin block propagation. Our data does not identify a limit on scaling when compact block and xthin block propagation protocols are deployed.

Further simulations of 10MB compact blocks produced results consistent with our 0% to 23.1% confidence interval for the orphan rate. An orphan rate of 10% would make an economic analysis of block size pertinent. Xthin block propagation with an orphan rate under 1% suggests that there would be no economic limitation on block size up to 10MB.

3 Further Work

Further work to be done may include simulations which involve larger blocks. It would also be instructive to develop independent testnets to emulate the network. We would also like to understand how the graphene protocol [OAB+17] effects block propagation.

The authors would like to thank Jeremy Liu for helpful discussions and review of the manuscript. The simulations were carried out at Arizona State University's Center for Assured and Scalable Data Engineering (CASCADE). This research was made possible in part by a grant from the Dash network.

References

[Cha18] Nakul Chawla. Dash Simulator. `https://github.com/thenakulchawla/dash-simulator`, 2018.

[Cor16] Matt Corallo. BIP-0152, Compact Block Relay. `https://github.com/bitcoin/bips/blob/master/bip-0152.mediawiki`, 2016.

[GKW+16] Arthur Gervais, Ghassan O. Karame, Karl Wüst, Vasileios Glykantzis, Hubert Ritzdorf, and Srdjan Capkun. On the Security and Performance of Proof of Work Blockchains. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 3–16, New York, NY, USA, 2016. ACM.

[Nak08] Satoshi Nakamoto. Bitcoin a Peer-to-Peer Electronic Cash System. `https://nakamotoinstitute.org/static/docs/bitcoin.pdf`, 2008.

[OAB+17] A. Pinar Ozisik, Gavin Andresen, George Bissias, Amir Houmansadr, and Brian Neil Levine. Graphene: A New Protocol for Block Propagation Using Set Reconciliation. In *Proc. of International Workshop on Cryptocurrencies and Blockchain Technology (ESORICS Workshop)*, September 2017.

[Tsc16] Peter Tschipper. BUIP-010 Xtreme Thinblocks. `https://github.com/BitcoinUnlimited/BUIP/blob/master/010.mediawiki`, 2016.