Bitcoin Virtual Gold:
Bitcoin with Proof of Transaction (PoT)

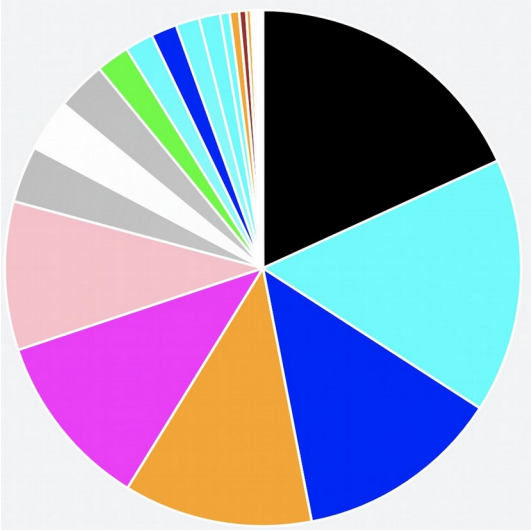
NullFunctor
github.com/BitcoinVG

Abstract. Satoshi Nakamoto's original vision for Bitcoin was to create a peer-to-peer version of electronic cash. The majority of successful forks of the protocol try to improve on this vision further and provide a more scalable and efficient system for payments. We see Bitcoin's greatest promise not as a medium of exchange but as a store of value – a better form of gold, not cash. We propose a set of modifications to the original protocol aimed at fulfilling this promise by creating the ultimate electronic store of value. By increasing Bitcoin's mining decentralization, we are able to tackle Bitcoin's greatest flaw as a form of gold – mining centralization

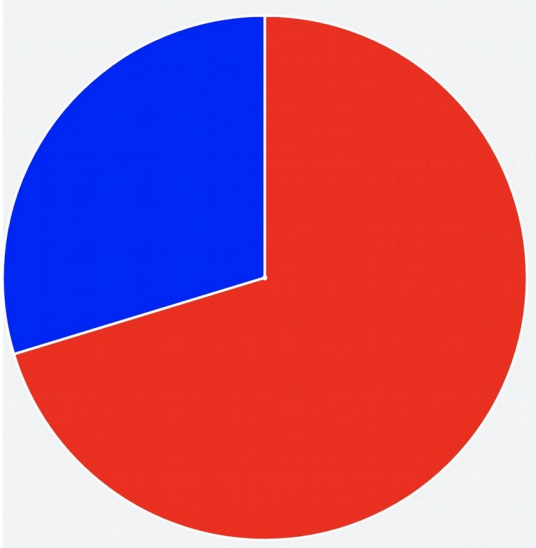
1. Bitcoin's Centralization Problem

Bitcoin was originally designed to be decentralized, through its journey it didn't end up following this path and never will. Some say because Bitcoin has become centralized that it is doomed and is on a death spiral, however give the world a large enough incentive and decentralization can surface back to life bringing the new Bitcoin back to Satoshi's original vision of staying decentralized.

Pictures are worth a thousand words, take a look at the charts below. (Keep in mind that 51% of the hashing assumes the power to do something) The chart below shows the distribution of hashing power for Bitcoin (Mid 2018). Technically, one needs to hijack 3 to 4 entities to gain control of the hashing power. Let's pretend AntPool, BTC.com, ViaBTC, F2pool, BTCtop are independent non-colluding organizations and not just one entity hiding to be 5. GHash failed because they publicly demonstrated owning 51%. Chinese miners have learned from this and are smarter than publicly showing a 51% ownership. The chart below shows how nicely the hash rate is divided with no clear leader. The distribution is still heavily centralized due to the accumulation of the 4 largest slices of the pie forming a centralized entity with over 51% hashing power. In general countries have the authority over miners residing in their country.



Here is what happens when we compare China vs. non-China with regards to Bitcoin hashing power.



Clearly this is a threat to the Bitcoin community. Over 51% of the hashing power resides in China.

Even just looking at the distribution of miners and assuming they are independent, it is clear that just a few Bitcoin nodes are responsible for making decisions on how to form the blockchain.

2. How Bitcoin Determines the Main Chain

Bitcoin has an abundance of code that determines how an incoming block gets processed and ultimately finds its way into a tree like structure. The block index is a data structure that allows blocks to chain together to form many different chains. Why do we need many chains? Well, this can happy under normal conditionals too. If two blocks are found around the same time they start to propagate into the Bitcoin network. The network is live and has delays in it. One node that has a ‘best block’ might be different than another node that has a ‘best block’. Usually the best block will be resolved as the next block is mined and its work metric becomes larger than the other chain.

We are not here to dissect the many different corner cases that Bitcoin must account for but rather from a high level. So what is it that really determines if a block can be connected to the main chain and be considered at the tip? Ultimately it comes down to the amount of work on chain that determines if the best chain was formed. There are other considerations like deltas with block times, stale chains, but in a simple way, the amount of work on the main chain determines the best chain. If a block hash gets lucky and gets more zeros out in front, than the previous block, this does not mean more work, but rather that the miner was just lucky. All blocks for a given difficulty level will receive the same metric for the amount of work done.

Since it is the amount of work that determines the best chain, who controls this? The control loop in the code adapts to try and make the deltas between each block at a 10 minute interval. As more mining power comes on the network, the difficulty level increases and the consensus updates to expect more leading zeros in the front of the hash to be accepted. The amount of work is driven by the amount of mining hash power on the network. This means that a large mining company can start mining in secrecy and form a longer chain than the public chain and then attach the secret chain back onto the public chain. This will force the public chain to updates to what was the secret chain. This is called the 51% attack. It was pulled off with pure mining power. Nothing else was taken into consideration. What? Why wouldn’t the code take into consideration the amount of money that resided in that block? Clearly the more money in the block should obtain a higher logical metric for importance, right? This is where “Proof of Transaction” comes into play.

3. Proof of Transaction (PoT)

Proof of Transaction (PoT)is a way for the embedding the amount of coin that was processed in a particular block. Well, isn’t that already in the block? Yes, but it is in the block transactions and not in the Bitcoin header. Bitcoin only looks at the amount of work that was used to create the hash, namely the nBits. We need a way to bring the coin

volume from the block transactions and into the header so that Bitcoin's normal decision making process can still occur.

With PoT, the amount of coin volume gets embedded in the last 8 bits of the block's hash. For example, if there was almost no coin in a particular block, the block's hash would need to have all zeros for the last 8 bits. This can be seen in the following below.

```
0000000000546a7395c15302482e154a05cf11ef0659465c28266fb73da1dd00
```

The 10 leading hex digit zeroes are for the PoW, while the 2 far right hex digits ('00' in this example) feed into the PoT calculation. This particular block was a block that was mined with very little transaction volume, thus was '00'.

Creation of a hash is slightly altered from that of Bitcoin. Once a PoW difficulty level is met, it is passed to the node for verification before being added to the Block index. However, in PoT, there is a high chance that the block only met the PoW criteria and the PoT criteria wasn't met. For instance, if the block had a small transaction that should result in '00' at the top of the hash, the block would keep getting rejected until both the PoW and the PoT meets the criteria. Currently most ASICs only calculate and search for the PoW, the block would then get passed up to the node where the 2nd later verification of PoT would be checked. As PoT becomes more popular, ASICs would be modified to not pass up a block to the node until PoW and PoT thresholds are met. This would make for much more efficient processing.

We now explain how the PoT feeds into the overall "Proof" metric. BVG has a PoT metric system that goes from 0 to 255, 0x to 0xFF. Transactions between 0 BVG and 25500 BVG get compressed from 0 to 0xFF. Anything above 25500 BVG maxes out at 0xFF. Just like PoW block creation, PoT has criteria that must be met. In order for the consensus to accept a block, the amount of transaction volume in the blocks transaction vouts, must match the last 8 bits in the block's hash after decompression. The original PoW target calculates as usual and then the PoT metric is pulled out, brought back to 0 to 25500 and then the PoW metric is modified based on how much coin flowed through the block. A large transaction of over 25000 BVG would result in the original metric being made 25000 harder than before! This means that it would take hundreds of PoW only blocks to compete with one block that held a large transaction. This means that the attacker needs to try to perform a 51% attack by not only mining very fast, but also by putting large amounts of his own coin in each block if he wants to have any chance of succeeding with a 51%. The last statement above is what really makes the difference between PoW and PoT. BitcoinVG combines PoW and PoT together to form the 'work' metric.

Since the PoT metric magnifies the PoW metric, trying to 51% attack BVG by merely mining blocks without or with smaller transaction amounts would always fail if every block had at least a little more coin in each block than the attacker's blocks.

Let's apply BVG's PoT algorithm to modern day Bitcoin. Say BTC is $15,000 per coin and the average amount of BTC that flows in each block is around 2500 BTC. This means that if an entity wants to try to attack BTC, they must create a private network, keep sending at least 2500 BTC in every block and mine more blocks than the public network before attaching back to the public network.

What are the benefits from the above modern day BTC example? It is still possible to 51% attack both BTC and BTC with PoT, but theoretically, the existing BTC network can still be attacked by an entity with small amounts of BTC and being very lucky. Attacking with the PoT added reduces the number of groups that could pull off a 51% attack because now much more money is needed; $37,000,000 per block is needed per block to try and attack using PoT, only $0 and a boatload of ASICs are needed for attacking the current BTC blockchain.

We have seen that there is an additional benefit to adding PoT to the metric calculation. This paper will be submitted to the Bitcoin Core team as a possible Bitcoin Improvement Proposal (BIP). No matter the outcome, BitcoinVG's support of the PoT metric makes for a more protected Bitcoin from potential 51% attackers.

Thanks,
NullFunctor
Bitcoin Virtual Gold founder