A change in the mining and reward mechanisms in crypto-networks can drastically reduce global electricity consumption

© Giuseppe Gori - May 31, 2016 - gori@exacom.us

Abstract

Miners in distributed crypto-networks use specialized processors to validate and develop blocks of information that is maintained in nodes across the globe.
In this document we show how a change in the mining and reward mechanisms can drastically reduce the miners' hours of operation, and consequently their electricity consumption, which will translate into higher incomes. This is achieved without changing the proof-of-work concept and maintaining the current block chain mechanism, processing blocks at the same speed, and using essentially the same rules and processing resources.

Introduction:

The "mining" process of current block chain-based crypto-currencies is based on several ideas and techniques. Protocols have been implemented by various companies (e.g.: Bitcoin, Ethereum, Litecoin, etc.) based on the initial white paper by the fictional ***Satoshi Nakamoto***[i].

More details can be found in the Bitcoin Developer's Guide[ii], the Bitcoin Developer's Reference[iii] and the Bitcoin Protocol documentation[iv].

The block chain protocol was designed to work with any number of participating nodes (above a practical minimum). In the current crypto-networks, the nodes competing for the solution, producing "proof-of-work" and mining the next bonus coins, are **all** the peer miner nodes for that particular crypto-network (e.g.: Bitcoin).

For the proof-of-work mechanism to be effective in preventing malicious attacks, it has to consume a certain amount of computing resources and time. This time must be, in average, much higher than the time required to ask and receive a block of information from peer crypto-network nodes. The expected average time can be increased, when necessary, by increasing the difficulty of proof-of-work hash rules.

In this proposal, transactions continue to be propagated on the crypto-network for anyone to see by all miner nodes. No change is proposed to the propagation of the block chain.

Our objective:

Currently all miners' processors are constantly powered-UP using electricity, but doing mostly proof-of-work hashes, which more than 99.99% of the times are tossed away.

Our main objective is to drastically reduce[v] the number of the processors that need to be always powered-UP: For almost every new block iteration, a subset of them can do the work and guarantee the security and reliability of the block chain.

A secondary objective is to increase the profitability of miners, by reducing their costs.

The work of a miner processor essentially consists in:

- collecting, verifying and assembling transactions into a block, and
- using hashing techniques to assemble a “proof-of-work”, by looking for the shortest hash of the block.

While the first is “useful work”, the second is necessary and very expensive, but largely fruitless work. It was devised to deter malicious attacks by making this task very difficult to compute. However, many Kilowatts/hour are spent globally to accomplish this task.

Whether a mining organization has many specialized processors, or an individual miner has a single processor, each node can maintain an always active pre-processor whose functions are to request, receive and verify the latest transactions and check whether work needs to be done by the production “hashing” processor(s).

These production processors can then be normally powered OFF, until the pre-processor calls them into action (e.g.: Wake-on-LAN option). This event will happen, for example, once every sixteen blocks or more, depending on the length chosen for our period “P” (See “***The Period***” section, below).

For example: Let’s assume that, in a certain crypto-network, there are 100,000 crypto-network processors mining at all times. Our mechanism would keep, in the example, 1,000 of them working all the time (but not always the same 1,000) while the other 99,000 would be woken up to do work, for example, once every 16 blocks (in the Bitcom network, about two hours and forty minutes). In this example, the miners’ electricity consumption (and their global electricity demand) could be reduced to **less than one tenth** of the current consumption.

***Our strategy*:**

Any strategy that would enforce ***limitations*** on the number of miners based on their “identity”, such as their IP address, wallet or crypto-network address, is not workable, as miners can create new identities at will. It is part of the “anonymity” philosophy of crypto-networks.

However, if a miner is ***rewarded*** (in one or more ways, and “privileged”) when it maintains its identity, then we can count on the large majority of them to voluntarily do so, in order to maintain their privileges, for a certain period of time.

In our proposal, the “identity” of a miner (e.g.: a node number, a signature, a public key, a wallet address, a network or crypto-network address) included when a “privileged” miner propagates data to its peers, must correspond to an easily verifiable, already checked identity in the block chain.

For this purpose we can count on, and refer to, a previously verified identity: the one included with the generation (or coinbase) transaction introduced, at a previous time, in the block chain by each miner that has been a “solver” in the past.

A reference to this confirmed identity, by a miner hoping to earn more “coins” with the same identity, will be sufficient confirmation. No other miner would have interest in assigning their last block “coins” to another miner’s identity.

Accordingly, we propose the following changes in the mining mechanism and in the rewards structure.

The Period:

A "***period***" is established as a function of the block number. In our example, a period "P = 16" is established, where all miners work on the latest block only when its block number is divisible by 16. This is called an "***open block***": open for all miners to work on. The other blocks are called "***privileged blocks***".

Two groups:

We can envision the total number "T" of peer miners divided into two groups:

a) a verifiable group of "N" privileged miners (in our example, 1,000), and

b) the group of all other miners "T-N", (in our example 99,000).

The the claim to belong to the privileged group is simply verifiable by all peer miners. This claim is indicated, by a privileged miner, by including in the block being created a reference to its previously verified identity within the last "N" open blocks (in our example 1,000).

In order to simplify the verification process, this block number (a "reference link" to a previous generation transaction) can be included in the header of each block they create, as this is passed along to their peers, for verification purposes. No "hashing" of this field would be required.

This strategy would allow a previous "solver" miner to claim membership in the privileged group for about (N x P x m) minutes, where "P" is the period and "m" is the average amount of minutes required for a new block to be included in the block chain (in our example, for Bitcoin, a solver would remain in the privileged group for at least 44 days).

The privileged group can do the work currently done by all miners, because the number "N" can be big enough to satisfy minimum requirements.

The privileged group of miners have shown their proof-of-work and will be further rewarded for continuing to do work.

The probability of being able to claim membership in the selected group is the same for all miners, since all miners have an equal probability (factoring in the processor speed) to make a block and provide proof-of-work.

The probability $(m/M)_s$ of "m" malicious attackers out of the total "M" malicious attackers being members of the selected group, is the same as "M_t", the probability of "M" malicious attackers in the total group of miners.

Thus, in a privileged group above a practical minimum, the probability of a successful malicious attack by a group of malicious attackers is the same as for the total group of miners in current implementations.

Time to do the work:

The block chain would be propagated as usual, to all peer miners, for each block (privileged and open).

Group a) “privileged” miners

- Miners in the privileged group would be known to their peers because of their verifiable “reference” to a previous generation transaction.
- Miners in the privileged group would work on the block chain for every block, for a period of at least “N x P” blocks (in our example, 16,000 blocks). Each miner in the privileged group would compete for some reward **for every block** (See next Section).
- However, when the last block of the block chain is an “open block” (i.e.: when all miners are competing at the same time), they compete with the same rules and for the same rewards as all of their peers: Mined coins, transaction fees, and their privilege to be part of (or remain for a longer time in) the “privileged group”.
- Their rewards for verifying and solving “privileged blocks” would have to be re-adjusted (See next section).
- The difficulty required for the block hash for “privileged blocks” would have to be re-adjusted as well (i.e.: the hash may not need to be so short), in order for blocks to be built at approximately the current rate.

Group b) miners

- The rest of the miner community would all work on the block chain once every period ”P”.
- All miners’ hashing processors would then be woken up by their pre-processors once every “P” blocks (in our example, 16). Every miner’s reward for verifying and solving “open blocks” would have to be re-adjusted (See next section).
- Miners pre-processors would keep track of the latest blocks, verifying and collecting them, and waiting for open blocks, in order to wake up the hashing processors.
- Miner pre-processors would also accept and verify blocks coming from privileged miners. They would reject data in privileged blocks that is not coming from a privileged miner (i.e.: when the reference link to their confirmed identity is invalid).
- The difficulty required for “open blocks” should not have to be re-adjusted.

The rewards:

Group b) miners would “mine” only every “P” blocks. Their reward probability would be “P” times smaller, thus their reward coins should be increased (for example multiplied by “P”, or a large fraction of “P”). This would guarantee a monthly revenue similar to their current earnings, but a higher income, since their electricity expenses would be proportionally smaller.

As a result, more miners would be attracted to do this work and they may in average invest more of their money into faster processors.

Miners who are part of a “pool” that use a single “identity” to create their generation transactions and cash their rewards, would probably often belong to the privileged group.

Group a) “privileged” miners would work on every block (including “open blocks”). When working on “open blocks” they would compete with everyone else, and with the same (common) reward structure.

When working on “privileged blocks” (“P-1” out of “P” times) they would have a much higher chance of being the solver (in our example, 100 times higher probability). Thus their rewards for winning privileged blocks should be reduced drastically, but their reward should remain high enough for them to voluntarily continue to work. For example, they would continue to collect transaction fees.

Gradual introduction:

The flexibility in choosing the rewards, and the variables “N” and “P” allows for a gradual introduction of our mechanism, thus giving crypto-networks some time to adjust the above parameters, and evaluate the consequences (advantages and disadvantages).

With a period length of only 4, the energy consumption could possibly be reduced to about one third of the current average miner consumption, thus motivating most miners to adopt a pre-processor-based node architecture.

With a period length of 12 to 16, the energy consumption could possibly be reduced to less than one tenth of the current average miner consumption.

The rewards in transaction fees paid out for each block, would remain. These, collected by a privileged miner, should be a high enough incentive by themselves, providing crypto-networks with some of the “scalability” they are looking for, in the future.

By drastically reducing the energy costs, more miners may find it cost effective in participating in crypto-networks with more processing resources. As a result of such possible efficiency improvements, the difficulty of the standard block hash may have to be increased.

An initialization time, for the mechanism to go into effect would be required, in order to allow for the privileged group to build-up. This time would be proportional to “N” and “P”.

Off-peak periods:

In areas with high electricity costs, having implemented a pre-processor-based node architecture, some miners may be interested in further reducing their electricity costs by powering-up their processors only during “off-peak” periods[vi]. When the period “P” is relatively long, this technique should give them a chance to compete effectively with the rest of their peers.

Conclusion:

This document shows how crypto-networks can reduce global electricity consumption, by implementing a simple mechanism, which will allow miners’ specialized processors to stay asleep most of the time, and wake-up only when they are required. As a by-product, more people may find it cost effective to participate in crypto-network “mining” work.

[i] White paper from Bitcoin at: https://bitcoin.org/bitcoin.pdf

[ii] Bitcoin Developer’s Guide at: https://bitcoin.org/en/developer-guide

[iii] Bitcoin Developer’s Reference at: https://bitcoin.org/en/developer-reference

[iv] Bitcoin Protocol documentation at: https://en.bitcoin.it/wiki/Protocol_documentation

[v] This is done without reducing the network resiliency against coordinated malicious attacks from large numbers of nodes.

[vi] Periods of the day when their electricity cost is lower.