Sigmoid: ASIC-resistant Mineable ERC-20 Token

John Weligon
Electrical and Computer Engineering Dept.
sigmoid.token@gmail.com

I. Introduction

The advent of gigantic mining pools has long been a headache for a cryptocurrency field. Due to the possibility of 51% attack and block withholding attack, the very existence of the gigantic mining pools was one of the biggest threats to the security of proof-of-work cryptocurrencies. Not only in terms of security but also in terms of energy efficiency were those mining pools a nuisance. It is known that the process of creating Bitcoin to spend or trade consumes around 91 terawatt-hours of electricity annually, more than is used by Finland, a nation of about 5.5 million.

Various technical attempts have been made to solve this problem. One of them was to abandon proof-of-work consensus mechanism and switch to proof-of-stake mechanism. Ethereum, which Sigmoid is going to be based on, is an example of cryptocurrency that is planning a switch from PoW to PoS mechanism. Though this changeover of the consensus mechanism can effectively alleviate the aforementioned issues, to miners can the whole changeover be seen as 'being thrown away like an old shoe.'

To miners, mineable ERC-20 tokens like 0xBitcoin were great substitutes for cryptocurrencies preparing for a changeover. The move was faster than expected and the competition between miners is already fierce. In the case of 0xBitcoin, the hash rate is already enormous 100Th/s. This is because there was an influx of ASIC miners from Bitcoin. Though 0xBitcoin doesn't have to worry about the danger of 51% attack, since it is a token on Ethereum chain, ASIC-friendliness of 0xBitcoin is certainly a disaster to Ethereum miners who used to run a rig with a few GPUs attached.

Therefore, we bring another older attempt for decentralization: to use an ASIC-resistant hashing algorithm for proof of work. For example, Scrypt and Ethash, which were known to be GPU-friendly and ASIC-resistant, were respectively selected as a hashing algorithm by Litecoin and Ethereum. Those memory-hard hashing algorithms demonstrated their effectiveness for a long period though ASIC modules for those algorithms have now been developed and commercialized.

In this paper, a new ASIC-resistant ERC-20 token with a relatively new memory-hard hashing algorithm, **Sigmoid** is presented. For minting, Sigmoid uses Balloon hashing for its key derivation function, which is a NIST-recommended algorithm theoretically proven to be memory-hard. In this way, we expect to achieve thorough decentralization and bring about short-term energy-efficiency and equality through incapacitation of ASIC miners.

II. Rewarding and Difficulty Adjustment System

All parts of the Sigmoid system other than hashing algorithm follow those of 0xBitcoin. Just like 0xBitcoin, maximum total supply of Sigmoid is 21 million tokens and the amount of SIG issued per block is set to decrease logarithmically, having a 50% reduction every time half of the remaining supply has been mined. In result, total supply of Sigmoid will never exceed 21 million.

The rate of block creation is adjusted every 1024 blocks to aim for 1 SIG minting per 60 ETH blocks, which is roughly 6 per hour. The contract increments the reward era if the tokens minted count has exceeded the maximum era supply which is calculated by Eq. (1). However, all difficulty targets are bound within minimum and maximum difficulties of 216 and 2234 respectively.

$$\text{max era supply} = \text{total supply} - \frac{\text{total supply}}{2^{\text{reward era}+1}} \tag{1}$$

There will be 40 reward eras until the mining will halt. [1]

III. Hashing algorithm

Sigmoid uses Balloon hashing for its key derivation function. Balloon hashing is a NIST-recommended key derivation algorithm which is theoretically proven to be memory-hard. Memory-hardness of this particular key derivation function enables Sigmoid to be ASIC-resistant. [2]

Another strength of Balloon hashing other than memory-hardness is its versatility. Since Balloon hashing is only a key derivation algorithm, it can use almost every hash function for its hashing / pseudo-random-number generation processes. By using hash functions that are already built in Solidity API, it is possible to reduce the gas consumed for validation of mining results. In the case of Sigmoid, the hash function is SHA-256.

Like 0xBitcoin, to prevent pre-mining and Man-in-the-Middle attack, the digest of Sigmoid's hash function includes a recent Ethereum block hash and msg.sender's address. In addition, the digest also includes another variable named *cnt* to secure memory-hardness. *cnt* increases gradually as the hashing process continues and helps mixing the hash blocks.

References

[1] https://github.com/0xbitcoin/white-paper-v2
[2] Boneh, Dan, Henry Corrigan-Gibbs, and Stuart Schechter. "Balloon hashing: A memory-hard function providing provable protection against sequential attacks." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2016.

***Index Terms*—ASIC-resistant, mineable, ERC-20, Ethereum, PoW**