Proposal for the Merit Caching Consensus Mechanism

Luke Parker

January 25, 2019

Abstract

Proof of Work requires both the current state and extended periods of time to function, which is unacceptable for instant transactions on a cryptocurrency network. If a miner only has to sign a transaction, effectively instant transactions can be achieved. The issue then becomes a lack of reputable verification, as effectively random transaction signatures aren't anymore secure than just removing mining altogether, opening the network up to double spends. Also, these signatures don't declare to which addresses new coins should go.

By using Proof of Work to generate Merit, we can cache work, removing the need for the current state. This is accomplished via a separate Blockchain that would be mined to generate Merit. This non-transferable Merit would approve transactions on a form of Directed Acyclic Graph. As signing can be done almost instantly, this system allows for the security of Proof of Work to be applied to an effectively instant transaction system.

Contents

1 Employed Data Structures

Merit Caching depends on a dual-structure database, with one being a blockchain and the other being an asynchronous structure, such as a Directed Acyclic Graph or Lattice (as used in this paper). This is due to the Lattice being needed for instant transactions, yet a time-proven, secure, and reliable blockchain is the mineable structure. If we mined the Lattice on a transaction-by-transaction level, we'd have to wait to generate the work, as work's security is provided by the time needed to calculate it. Therefore transactions wouldn't be instant.

1.1 Blockchain

A blockchain is linear chunks of data (blocks), or a linked list. No existing blocks can be edited and all new blocks always appear at the end, defining the 'x-axis' of the database as time. This means someone who downloads the blockchain years after its creation can see how it was formed over time and verify its state.

Merit Caching considers the blockchain with the most accumulated work valid, just as most Proof of Work systems do. The greater work represents greater processing power and the chain is therefore the most secure. New blocks are added only if miners can generate a hash which beats the difficulty, a property that exists to require high amounts of work, and is generated based on how often blocks are added. This also allows semi-consistent spacing of blocks.

It should be noted it is possible to not use Proof of Work, but instead use Proof of Stake. However, that creates concerns around the distribution of Merit.

1.1.1 Blocks

In blockchain-based cryptocurrencies, blocks contain transactions. In the Merit Caching mechanism, blocks contain a list of Verifications that reference transactions from the Lattice. By knowing every verified transaction, we provide a master overview of the Lattice to be used for syncing and deciding block rewards.

1.2 Lattice

A Lattice, or Block Lattice, is a modified form of a Directed Acyclic Graph. Each address has its own Account (blockchain), and only that address can write to it. This allows definitive Entry (block) ordering, even at high speeds with severe network lag, asynchronous writing to the database (if the addresses are different), and easy comprehension by humans.

1.2.1 Transactions

Transactions are composed of two parts: Send and Receive entries. When a user sends cryptocurrency to another user, the sender creates a Send Entry. For this data to be added to the receiver's Account safely, as rapid incoming transactions could overwrite the indexes with which the user is trying to send cryptocurrency, and therefore stop the user from spending money, the receiver must create a Receive Entry that declares the index.

1.2.2 Verifications

Verifications are created by Merit Holders, and include a transaction hash. Their purpose is covered in 2.2.

2 Mining and Verifying Transactions

2.1 Mining

Merit Caching involves mining the Blockchain via a Proof of Work algorithm, and any Proof of Work algorithm, from SHA256 to Keccak256 to Scrypt to Lyra2, satisfies Merit Caching's needs. Successfully mining a block generates Merit, a non-transferable value that decays over time. This is because Merit mined years ago should not be in play when the profit generated is so small it's ten decimals down and clogging the bandwidth with its Verifications.

2.2 Verifying Transactions

When a new Entry is added to the Lattice, everyone with Merit can sign it or not sign it. Anyone who signs the Entry places their Merit behind it. A transaction only becomes valid when it has 50+% of all Merit behind it. This means if a double spend attempt is broadcast, and both attempts are at 49.9% because they were sent out at the same time to different parts of the network, the person with the last 0.2% will only sign one, stopping the double spend before it's confirmed. If a malicious Merit Holder signs a double spend, that Merit Holder will lose all their Merit for this action. The Merit behind the transactions will then be recalculated, and one may need to cross the 50% threshold again.

In order to remove orphans from the equation, or positions where all created Entries never gained 50%, it is possible to consider the Entry with the most Merit behind it the verified Entry, after said Entry's Epoch (see 2.3).

2.3 Incentive

Merit Holders earn coins from the Blockchain after an Epoch. An Epoch is a group of blocks that start on a transactions first appearance on the Blockchain and end a fixed amount of blocks later. The purpose of the Epoch is to allow for all Verifications to be received and mined. At the end of a Epoch, Merit Holders get coins based on:

- Percent of Merit they placed behind transactions;
- Percent of verified transactions in this Epoch they helped verify.

3 Attack Vectors

Merit is earned by mining the blockchain and used to verify transactions on a Lattice. These Verifications feed back into the Blockchain. As the Blockchain provides a conclusive and final overview, Merit Caching successfully avoids most attack vectors found on a Proof of Stake system, yet faces the same vectors found with Proof of Work blockchain systems, with a bit of fault tolerance.

3.1 Double Spends

As covered above, double spends can be submitted, yet only one will be verified without a malicious Merit Holder. When a Merit holder is malicious, they lose their Merit, reverting the transactions's states so either both transactions are pending or only one transaction is verified. This would be resolved in a matter of seconds, and does have partial premature detection as Nodes can see if the potential transactions are both close to 50%, and wait seconds once one was verified to see if the other one also becomes 'verified'. When there's a high-value transaction, and a service wants to be absolutely sure the transaction was verified, the service can wait for the epoch to end. That said, the service only really has to look for other potentials, check if the potentials have relatively equal amounts of Merit behind them, and wait a few seconds for the double spend to 'succeed', before it's flagged/resolved.

3.2 51% Attack

As the chain follows the most-work rule, the chain can be rewound if an attacker has 51% of the mining power. Since Verifications are based on weight created by the Blockchain, editing the Blockchain alters transactions on a mass scale, potentially not just reverting transactions to pending but invalidating them entirely. That said, a lot of blocks

must be reverted if an attacker wants to change the Merit balances significantly, and after the chain is rewound, the data from the Lattice can be resubmitted. If no conflicting data was entered in the meantime, which requires 51% of the Merit (not the mining power), everything will go back to normal.

This attack changes on an orphan-less chain, as by mining just seven blocks, it's possible to remove and replace an existing verified transaction without any Merit accumulated in advance. By adding a six block checkpoint system, this specific attack, and all 51% attacks, can be avoided.

3.3 51% Attack Part Two

Just as Blockchain miners who get 51% of the mining power can selectively archive Verifications, causing orphaned transactions and manipulated payouts, anyone who gets 51% of the Merit can selectively verify transactions. That said, this would require mining 51% of blocks for a unreasonably long time.