Pareme: The Proof Of Work Blockchain

Alejandro Ochoa
www.pareme.org

Abstract. We propose Pareme, a lightweight blockchain secured by proof of work (PoW). It separates application logic from verification, allowing services and applications to outsource verification to Pareme's PoW system. Each block contains proof-of-work data and a custom hash chosen by the miner. Pareme has no native tokens, coins, public/private keys, or block ownership. This design makes it the lightest blockchain in use. Without block ownership, targeted attacks are minimized, enhancing decentralization. Miners select the hash, representing any content they choose. Potential use cases are explored later in the paper.

Introduction

Problem
Blockchains that handle data or smart contracts grow large quickly. Many systems lean on each other for security (e.g., Bitcoin-Lightning Network or Litecoin-Dogecoin relationships) by embedding data into the other. This works because PoW is secure, but utilizing Bitcoin, for example, as a storage for data not essential to the system itself bloats it and creates vulnerabilities. The core issue is simple. How can content be secured on an open platform without the mass data bloating its size?

Solution
The approach is to have a dedicated blockchain that stores and secures hashes of the content that is to be protected. A hash maps any content to a fixed size. This simplifies storage and distribution. Notably, content cannot be reconstructed from its hash alone—a feature, not a flaw. It ensures privacy and enables Pareme's key function: verification. We elaborate below.

Securing Hashes To Protect the Content
The Bitcoin-Lightning Network and Litecoin-Dogecoin relationships are such that one network is embedding data onto the other to utilize its PoW security. To understand what's happening here, we realize that simply a hash needed for a system is being embedded into a larger PoW system to secure/confirm/verify content for the smaller system at the expense of the larger system. Pareme fills this role as a blockchain who's sole responsibility is to mine blocks which contains such hashes. This gives a place for smaller systems to utilize a larger shared PoW chain for its security.

A Theoretical Overview

A Standalone Proof-of-Work Chain For Unified Security and Verification Outsourcing
Proof of Work (PoW) secures Bitcoin effectively. While other PoW cryptocurrencies exist, they often fail to match Bitcoin's success. One reason is that splitting hash power across multiple chains is inefficient.

Each new chain demands separate mining efforts, though some leverage existing chains' algorithms via merge mining. In merge mining, miners embed a hash from a "Child" protocol into a "Parent" chain (e.g., Litecoin-Dogecoin). The Parent treats this as arbitrary data, while the Child relies on it for security. The

miner's task remains unchanged, only needing to include a given hash. However, multiple merge-mining setups with different currencies fragment the mining ecosystem and isolate hash power.

PoW can also secure non-currency systems. In physical domains, systems like different branches of militaries protect land, air, seas, and space, effectively outsourcing defense from individuals to larger systems allowing individuals to specialize without needing to expend defensive effort. In cyberspace, no such overarching defense exists; each application must independently manage its own security against threats like DDoS attacks, bots, and viruses. This is inefficient and leaves vulnerabilities. A standalone PoW Parent chain, like Pareme, allows applications to share hash power, creating a collective defense. By pooling resources, applications strengthen their security together, reducing the burden on individual systems.

Pareme is a standalone PoW blockchain that consolidates hash-power from all miners into a single chain. Non-dedicated miners can join, mining blocks with a user-specified hash for a fee. Third-party applications share a unified secure system through this chain. Miners focus solely on one chain, streamlining their efforts.

Decentralization and Mining Dynamics
Pareme has no block ownership or native rewards. Miners embed a chosen hash into the chain for their own purposes: verification, content security, or otherwise. Networks utilizing Pareme may have mining pools for their hashes to organize final payout of their native rewards. However, the absence of Pareme rewards renders mining pools for Pareme obsolete because miners gain only the inclusion of their desired hash, making independent mining more practical.

Security
In Pareme's architecture, a 51% attack does not imperil the blockchain itself but rather targets a specific connected system leveraging its PoW security. Without native tokens or transactions, Pareme's standalone integrity remains intact under majority hash power dominance, shifting the motive of such an attack toward an external application such as a merge-mined child chain or third-party verification system. However, any connected system benefits from the collective defense of Pareme's aggregated hash power. An attacker seeking to manipulate or censor a single application's hashes must overcome the total computational effort of all participants, not merely the subset tied to the target. This unified PoW structure ensures that individual systems inherit enhanced resilience, as the broader network's hash power acts as a fortified shield against isolated threats.

Comparisons to Existing Systems
Bitcoin stores data via its transactions, bloating its chain for a purpose it does not fit—many systems exploit its security to embed hashes, swelling it further. Ethereum and similar platforms carry smart contracts, adding complexity and size. Pareme strips these away, using proof of work only for hash verification. Its 80-byte blocks contrast with Bitcoin's megabyte-scale blocks, offering a leaner alternative for content validation. Pareme alleviates this burden on Bitcoin, providing a new location for hash-based systems.

A Technical Overview

Pareme is a compact blockchain, distinct from those supporting currency or smart contracts. Each block carries 80 bytes:

- **Proof Of Work Data**
  - Height: 4 bytes
  - Timestamp: 8 bytes
  - Previous Block Hash: 32 bytes

- Nonce: 4 bytes

- **Payload**
  - Content Hash: 32 bytes

Blocks target 10-second intervals, resulting in approximately 3 million blocks per year, or roughly 3 GB of data per decade.

Proof of Work & Difficulty Adjustment
A block's validity is determined by its difficulty value, adjusted every 2,016 blocks. A miner constructs a potential block with their chosen data, hashes it, and compares the result to the current difficulty. If the hash exceeds the difficulty, the nonce is altered, and the process repeats until a valid hash is found.

The difficulty is calculated as:

$$\text{New Difficulty} = (\text{Old Difficulty}) \times (\text{Ratio})$$

Where:

$$\text{Ratio} = \frac{\text{Average Time of Last 2016 Blocks}}{\text{Target Block Time (10 seconds)}}$$

With:

$$0.25 \leq \text{Ratio} \leq 4$$

Additional verification steps include:

- **Height and Previous Block Checks:** These ensure the referenced previous block is valid and the height value has incremented by exactly 1.

- **Timestamp Checks:** Ensures timestamp integrity, accounting for global time zones and propagation delays, with limits on future or past deviations.

Full Nodes & Pruning
Pareme's 80-byte blocks impose minimal storage demands. However, users may opt for a pruned version of the chain, where older blocks are discarded. Since each block references its predecessor, extremely old blocks are not required to verify new ones. At minimum, a node needs the last 2,016 blocks (~160 KB) to confirm a new block's difficulty and perform other verification steps.

While pruning is viable, retaining the full historical chain offers advantages. Certain use cases, detailed later, may require analyzing the frequency or existence of a specific hash across the entire chain. Nodes interested in such applications would thus benefit from maintaining the complete record. For basic verification and mining, however, a pruned node suffices.

Use Cases

Merge Mining
Other cryptocurrency projects can outsource their mining security to Pareme, similar to how Dogecoin uses Litecoin. Miners embed a Child chain's block hash into Pareme, securing both chains with one proof-of-work effort. Child chains gain Pareme's hash power, enhancing security and reducing costs

without fragmenting miners. Rewards come from the Child chain, while Pareme remains lightweight and efficient.

Access to Third Party Systems
Third-party systems can implement a verification mechanism for users by requiring their credentials to be hashed on-chain. This approach can serve as a simple defense against bots, as each connecting bot must be verified on Pareme.

Document Authenticity
Consider a document of disputed origin. Users mine the hash of the version they deem authentic. Observers tally blocks associated with each hash to identify the most trusted variant. For instance, Bitcoin users could signal their preferred Bitcoin software version by mining its hash, helping newcomers identify the crowd favorite. Observers can also compare recent blocks to the full historical record to gain insight into shifts in favorability over time.

Timestamped Records
A user hashes a file—say, a legal contract or artwork— to prove its existence at a given time. The user mines the hash into a block themselves or pays for mining power to do so. The blockchain's timestamp and hash serve as a public, tamper-proof record.

Ideology Voting
Users can embed hashes representing a set of principles or beliefs they support, effectively casting a vote for their preferred ideology in a global, decentralized space. For instance, an individual might hash a document outlining their stance on key issues—economic policies, social values, or technological priorities—and mine it into Pareme. Given different sets of ideologies, observers can tally which hashes of those sets have the most blocks, reflecting broader support. This creates a transparent, tamper-proof record of ideological alignment over time. Users could track shifts in sentiment by comparing recent blocks to historical data, offering a dynamic view of global preferences without centralized control.