I’m writing this posts for two reasons:

1. I feel like erasure coding is not discussed enough as a data availability solution for Plasma implementations
2. Yesterday, @vbuterin and @musalbas published a paper which proposes an interesting data availability scheme based on Reed-Solomon erasure coding. Their blockchain model uses SMTs (the state is represented as a key-value map), so it can support both UTXO and account-based chains.

I know erasure coding gives probabilistic data availability guarantees, but those can be really

high, as shown in the section 5.6 of the paper. Why Plasma folks are not considering this more often? Is it hard to implement? Something else?

Two main challenges of Plasma are ensuring correct transaction execution (preventing incorrect/malicious transactions) and ensuring data availability. It seems like SNARKs/STARKs should take care of the former in future, and erasure coding might be an elegant solution for the later (most implementations currently require users to download whole chains, which is quite a burden)?

Hope this can start a constructive discussion.

arxiv.org

[

](https://arxiv.org/pdf/1809.09044.pdf)

1809.09044.pdf

885.81 KB