TLDR

: We suggest an optimised block header design for proposer and attester signatures. The simple consensus-level definition makes best use of BLS aggregation without obstructing a network-level optimisation eliminating a roundtrip.

Construction

We require proposers to be attesters for their own proposals. The consensus-level definition of the block B_i

in a shard (or the beacon chain) is B_i=[U_i, A_i]

where:

- Unsigned proposal

: The “unsigned proposal” U_i

contains neither proposer nor attester signatures.

- Aggregate attestation

: The “aggregate attestation” A_i

is the BLS aggregation of “attestations” of U_i

, i.e. attester signatures of U_i

.

At period i

the network-level protocol goes as follows:

1. the proposer aggregates previously gossiped attestations for U_{i-1}

to construct A_{i-1}

1. the proposer gossips [U_i, A_{i-1}, P_i]

where P_i

is the proposer signature of U_i

1. attesters verify A_{i-1}

and P_i

(against U_{i-1}

and U_i

respectively) and gossip their attestation of U_i

Discussion

Notice the consensus-level definition of a block only has a single BLS signature (optimally verified with 2 pairings). It also does not “mix” components from different periods, i.e. the aggregate attestation A_i

is for U_i

, not U_{i-1}

.

Additionally, the network-level protocol is optimal in the sense that the proposer (and the attesters) gossip a single message. This is thanks to the network-level optimisation where the proposer includes the aggregate attestation of the previous unsigned proposal in his gossip message.