@JustinDrake has a proposal to use ring signatures to make a private-lookahead block proposerhere:

1. During epoch N-1, any validator can submit a linkable ring signature proving they are a validator into the chain
2. The ring signatures are randomly shuffled
3. During epoch N, the ring signatures are randomly shuffled, and validators can reveal the private data used to make the signature, thereby revealing that they created some given ring signature and so that they have the right to create a block at some given time

Here is an alternative that's purely hash-based. Suppose we have two hash functions, H1 and H2, eg: H1(x) = SHA3(0x01 + x)

, H2(x) = SHA3(0x02 + x)

. Assume the existence of functioning and highly efficient mixers (eg. coinjoin). When validators join, they are required to commit, using some mechanism using hash function H1 (possibly a Merkle tree), to a mapping i \rightarrow V_i

where V_i

is the secret number that they will need to reveal during epoch i

.

During epoch N-1, anyone can submit a value into the chain, along with a medium-sized deposit (eg. 100 times the staking reward); the intention is that they submit H2(V_N)

. The submitted values are shuffled. During epoch N, a validator can reveal that they have a right to create a block by submitting V_N

, along with a proof (eg. Merkle branch) for the commitment. V_N

can be checked against the previously submitted hash directly, and the commitment can be checked; both checks must pass for the block to be valid and for the validator to recover their deposit.

Note that even if the validator's block does not make it into the chain, the validator can get their deposit back at any point in the future by submitting a suitable V_N

and proof. It is possible for anyone to "clog up" the system by submitting invalid values, but this is expensive; the only non-money-losing strategy, aside from not participating, is to submit the single correct value for H2(V_N)

during epoch N-1.

I suspect it may be possible to create a ring sig alternative that's purely

hash-based, because we are dealing with an easier problem: it's ok for the link to be revealed during the second step, so it's more like a "ring hash" than a "ring signature"; will keep thinking more about this. Maybe there's something in the existing commitment scheme literature?