

Prerequisite: [Exponential epoch backoff](#)

Another possible set of rules for an exponential epoch backoff:

- If the last epoch finalized a checkpoint, the epoch length is $\max(50, \text{prev_epoch_length} / 2)$

blocks.

- Otherwise, the epoch length is double the length of the previous epoch.

Essentially, instead of setting the epoch length to 50 if we reach finality, we instead just halve it (with a base of 50 blocks). The benefit to this approach is that in the case of extended network latency, we reach finality in fewer blocks, on average, as compared to the first proposal.

For example, imagine that it currently takes 75 blocks for 2/3 of validators votes to be included in blocks for an epoch (due to network conditions). With the first proposal, it takes $50 + 100 + 200 = 350$ blocks to reach finality every time. With the above rules, it takes 350 blocks to get finality the first time, but then only 100 blocks (and then switches between 350 and 100), which is less on average.

The disadvantage of these rules is that, in the case of a transient network partition, it will take longer for the network to get back to the fastest possible finality time (50 blocks).

Depending on if we're more worried about recovering from a temporary network partition as quickly as possible or reaching finality faster (on average) in the case of extended network latency, one rule set might be better suited than the other.

P.S. As the first proposal doesn't change the safety proof, I don't think this does either