The current sharding spec has fixed-length periods of 5 blocks during which validators can add collation headers to the VMC. There's a tradeoff with period length:

- Nominal shard pace

: Smaller periods increase the nominal shard pace.

- Hit rate

: Larger periods increase the probability of validators hitting their periods.

The actual

shard pace is the product of nominal shard pace and hit rate, and is something we want to optimise for. The adversarial

hit rate is also important to optimise for security (to deal with e.g. main shard censorship, offchain DoS attacks, high network latency).

We suggest a collation proposal mechanism that relaxes the notion of period to improve actual shard pace and adversarial hit rate.

Construction

We call "strict periods" the old notion of fixed-length periods and build "loose periods" with two new rules:

1. Left extension

: If a header is added in its respective strict period the next validator is allowed to add the next header in that same strict period.

1. Right extension

: A validator that misses its respective strict period is allowed to add a header before the next header is added. In case conflicting headers are added in the same strict period, the fork choice rule gives precedence to the most recently selected validator.