A Finality Calculator for Filecoin's Expected Consensus

Guy Goren and Jorge M. Soares

Technical Report

PL-TechRep-2024-001

2024.02.01

A Finality Calculator for Filecoin's Expected Consensus

GUY GOREN and JORGE M. SOARES, Protocol Labs

We propose a finality calculator for Filecoin's Expected consensus that considers what takes place during epochs and can attain, under normal operating conditions, an error probability of 2^{-30} in 30 epochs (15 minutes) - a 30x improvement over the current 900-epoch threshold. It depends only on a node's local view and can be implemented without protocol changes.

CCS Concepts: • **Security and privacy → Distributed systems security**.

Additional Key Words and Phrases: blockchain, consensus, finality

1 INTRODUCTION

Filecoin's Expected Consensus (EC) comes with probabilistic finality and a 900-epoch soft finality threshold, intended to achieve a finality guarantee (tipset replacement probability) of 2^{-30}[4]. While network participants (e.g. exchanges, L2 operators, application developers) use different confirmation thresholds, they pessimistically wait between 100-900 epochs before considering a transaction final, leading to delays in the order of hours.

Instead of naively counting the number of epochs, we propose a finality calculator that considers what takes place during those epochs and, under expected operating conditions, can attain the same level of certainty in fewer epochs. We embark on an analysis of Filecoin's finality, i.e., the probabilistic guarantees that a given tipset will always be in the canonical chain, and show that, in real operating conditions, the same error probability 2^{-30} can be achieved in 30 epochs (15 minutes) – a 30x improvement.

This algorithm is practical, only requires visibility to blocks produced by honest miners, and can be implemented by clients or off-chain applications without requiring any changes to the protocol.

This document serves as a theoretical companion to FRC-XXXX[1]. Please refer to the FRC for more background, motivation, and implementation information.

TODO: Update FRC reference when available

2 PRELIMINARY CONSIDERATIONS

2.1 Randomness

Consider a chain with:

- N: number of validators
- f: adversarial fraction
- h: honest fraction $(1 - f)$
- e: expected number of blocks per round

Denote $X_f[r]$ the random variable that represents the number of blocks won by the adversary in round r. Similarly, $X_h[r]$ denotes the number of honest blocks. $\texttt{Bin}(k; n, p)$ is the Binomial distribution where n is the number of trials and p is the probability of success for each trial, k is the number of successes (value of a random variable). In our system, we have $n = N$ and $p = \frac{e}{N}$. Therefore:

$$\Pr[X_f = k] = \texttt{Bin}\left(k; n \cdot f, \frac{e}{n}\right) \tag{1}$$

2.2 Poisson approximation

We will repeatedly approximate the Binomial distribution by a Poisson distribution:

$$\texttt{Bin}(k; n, p) \approx \texttt{Pois}(k; n \cdot p). \tag{2}$$

[1] link to FRC

This approximation is good provided n is large and $n \cdot p \leq e \ll n$.

2.3 Communication

In this analysis, we assume the classic round-based synchronous communication model. Since the Filecoin documentation uses the term epochs, we will use rounds and epochs interchangeably. Moreover, for upper bounding the error probability, we assume consistent block broadcast, which has recently been implemented in the Filecoin network [2].

3 ANALYSING THE PROBABILITY OF ERRORS

Our analysis draws inspiration from techniques developed in [1] combined with techniques from [3], which we apply to the observed chain history of Filecoin. We denote by G the *good addition*, i.e. the number of blocks observed in the local heaviest chain (*lh-chain*) between target epoch s and current epoch c. We then split the analysis into three time spans:

Distant past The random variable L describes the adversarial lead at epoch s, i.e. the blocks produced by the adversary to form a competing chain minus the blocks observed in *lh-chain* up to the epoch s. L is non-negative: for adversarial competing chains that are lighter than the *lh-chain*, $L = 0$. When $L \geq G$, there is a possible safety violation.

Recent past The random variable B describes the blocks produced by the adversary between epoch s and the current epoch c. When $L + B \geq G$, there is a possible safety violation.

Future The random variable M describes the blocks expected to be produced by the adversary minus the number of blocks produced by honest validators when slowed by the adversary. When $L + B + M \geq G$, there is a possible safety violation.

Our analysis is based on the two lemmas below. Roughly, they establish that all chains that end with an honest block are visible to the user.

LEMMA 3.1. *Let b_h be a block produced by an honest validator at round r. Then the tipset chain ending at* `parent`(b_h) *is known to all honest validators by round $r + 1$.*

PROOF. Follows from the guarantees provided by Consistent Broadcast. □

LEMMA 3.2. *Let c be the current round. Let $t_a[v_i]$ be the "best" tipset chain of which validator v_i is aware (and would choose as parent), which ends in round a, and let $t_b[v_i]$ be the "best-competitor" tipset chain of which v_i is aware, which ends in round b. Then, in the interval $[b, c-1]$, the tipset chain $t_b[v_i]$ could have only been extended with malicious blocks (blocks proposed by malicious validators).*

PROOF. Assume that in the interval $[b, c-1]$, the tipset chain $t_b[v_i]$ has also been extended with honest blocks (blocks proposed by honest validators). Denote one of these (possibly single) blocks by b_h. By Lemma 3.1, the block b_h is visible to all honest validators at time c. Consequently, the chain ending at a tipset containing b_h is visible to all honest validators and to v_i in particular. Since this tipset chain extends $t_b[v_i]$, it is "better" than $t_b[v_i]$. This is in contradiction to $t_b[v_i]$ being the "best-competitor" tipset chain of which v_i is aware. □

Recall that the fork choice rule in Filecoin is based on chain validity and weight, where:

(1) Validity is determined by a set of rules that govern the correct construction of blocks
(2) Weight incorporates the number of blocks mined and a factor related to the storage in the power table

For simplicity, we ignore the subtleties of the two and refer to the "best" tipset as that at the end of the heaviest valid chain. We also conflate the weight of a chain with the number of blocks it contains. We can now start to derive the probabilities for each of the time spans.

3.1 Span 1: Distant past

Let s be the epoch for which the finality probability is being evaluated, and c be the current epoch ($c > s$). The random variable L describes the adversarial (secret) lead gained from the last final tipset (e.g. the tipset

at epoch $c - 900$) until epoch s. It behaves like a biased random walk whenever $L > 0$ but does not decrease when $L = 0$.

For each epoch $i \in [c - 900 + 1, s]$, the step expectation is $f \cdot e - chain[i]$, where $chain[i]$ is the number of blocks at the tipset of the *lh-chain* that was constructed at epoch i and $f \cdot e$ is the expected number of adversarial blocks at an epoch (i.i.d).

The fact that L cannot become negative, i.e., it "sticks to zero" changes the analysis somewhat since we cannot use the classic random walk model. Instead, to account for the distribution of L we can look at a reverse process (L') that starts at the tipset of interest of epoch s and moves backwards in time. In this case, we get that L' is distributed according to the following:

$$\Pr[L' = k] = \max\left\{\Pr[L'_1 = k_1], \Pr[L'_2 = k_2], ...\right\} \tag{3}$$

$$k_i = k + \sum_{j=s-i}^{s} chain[j]. \tag{4}$$

For each L'_i, we replace the binomial distributions of the steps by a Poisson distribution

$$L'_i \sim \texttt{Bin}\left(\sum_{j=s-i}^{s} f \cdot n, \frac{e}{n}\right) \sim \texttt{Pois}\left(\sum_{j=s-i}^{s} f \cdot e\right) \tag{5}$$

and get

$$\Pr[L = k] = \Pr[L' = k]. \tag{6}$$

3.2 Span 2: Recent past

The random variable B is independent from L and follows a simple binomial distribution as explained previously for X_f. For ease of computation, we again approximate the binomial distribution by a Poisson one:

$$B \sim \texttt{Bin}\left(\sum_{i=s+1}^{i=c} f \cdot n, \frac{e}{n}\right) \sim \texttt{Pois}\left(\sum_{i=s+1}^{i=c} f \cdot e\right) \tag{7}$$

3.3 Span 3: Future

The future production of honest blocks follows a binomial distribution. However, it might be that not all honest blocks are added to the same tipset, which happens when the adversary splits the honest chain. Specifically, to split the honest power, the adversary must be able to provide parent tipsets which are "better" than the currently available *lh-chain* one. We calculate a lower bound on the public chain growth rate based on the following two assumptions:

- Using the blocks of epoch i, the adversary can optimally split the network power for epoch $i + 1$.
- Only adversarial blocks from epoch i may be used to split the power at epoch $i + 1$.

The first assumption considerably favours the adversary, while the latter moderately favours us by somewhat limiting the adversary's capabilities. We conjecture that these assumptions correspond to a lower bound without them since, compared to without them, they seem to favour the adversary more than us. This is due to the practical difficulty of coordinating a perfect split that significantly benefits the adversary. On the other hand, using old blocks for the split has a limited effect due to their diminishing relevance.

With consistent broadcast, we have that, for $B[i-1]$ and $H[i]$ the number of adversarial blocks and honest blocks in epochs $i - 1$ and i, respectively, the honest chain grows by at least

$$Z[i] = \min\left\{\frac{H[i] + B[i-1]}{2^{B[i-1]}}, H[i]\right\}. \tag{8}$$

We therefore have that, at step i, the random variable M changes according to the sum $B[i] - Z[i]$. To simplify the calculations, we replace the random variable Z by Z':

$$\begin{aligned} Z' &\sim \texttt{Pois}(E[Z]) \\ &= \texttt{Pois}\left(E\left[\min\left\{\frac{H[i]+B[i-1]}{2^{B[i-1]}}, H[i]\right\}\right]\right) \\ &= \texttt{Pois}\left(\Pr(H[i]>0)\cdot E\left[\frac{H[i]+B[i-1]}{2^{B[i-1]}}\right]\right) \end{aligned} \tag{9}$$

We define the random process M_i recursively to be

$$\begin{aligned} M_i &\triangleq M_{i-1} + B[j] - Z', \quad M_0 = 0 \\ &= \sum_{j=1}^{i} B[j] - Z'[j] = \sum_{j=1}^{i} B[j] - \sum_{j=1}^{i} Z'[j] \end{aligned} \tag{10}$$

Moreover, for each $i \in \{1, ..., n\}$, we have that $\sum_{i=1}^{n} B[j] \sim \texttt{Pois}(n \cdot e \cdot f)$ and $\sum_{i=1}^{n} Z' \sim \texttt{Pois}(n \cdot E[Z])$. As the difference between two independent Poisson-distributed random variables, each M_i follows a Skellam distribution [5]. Thus, we conclude that:

$$M_i \sim \texttt{Skellam}(n \cdot e \cdot f, n \cdot E[Z]) \tag{11}$$

$$Pr(M = k) = \max\{Pr(M_1 = k), Pr(M_2 = k), ...\} \tag{12}$$

3.4 Error probability

For an observed good addition $G = k$, the safety violation event happens only if one of the three mutually exclusive events occurs:

(1) $L \geq k$
(2) $L < k$ but $L + B \geq k$
(3) $L + B < k$ but $L + B + M \geq k$

Knowing that

$$\Pr(L + B \geq k | L < k) = \sum_{l=0}^{k-1} \Pr(L = l) \cdot \Pr(B + l \geq k), \tag{13}$$

and that

$$\Pr(L + B + M \geq k | L + B < k) = \sum_{l=0}^{k-1} \sum_{b=0}^{k-l-1} \Pr(L = l) \cdot \Pr(B = b) \cdot \Pr(M \geq k - l - b). \tag{14}$$

We get

$$\begin{aligned} \Pr(error) &\leq \Pr(L \geq k) + \Pr(L + B \geq k | L < k) + \Pr(L + B + M \geq k | L + B < k) \\ &= \Pr(L \geq k) + \sum_{l=0}^{k-1} \Pr(L = l) \cdot \left(\Pr(B + l \geq k) + \sum_{b=0}^{k-l-1} \Pr(B = b) \cdot \Pr(M \geq k - l - b)\right) \end{aligned} \tag{15}$$

ACKNOWLEDGMENTS

We would like to thank all the community members who provided input and reviews for this work, including Alejandro Ranjal-Pedrosa, Irene Giacomelli, Juan Cianci, and Marko Vukolić.

REFERENCES

[1] Dongning Guo and Ling Ren. 2023. Bitcoin's Latency–Security Analysis Made Simple. In *Proceedings of the 4th ACM Conference on Advances in Financial Technologies* (Cambridge, MA, USA) *(AFT '22)*. Association for Computing Machinery, New York, NY, USA, 244–253. https://doi.org/10.1145/3558535.3559791

[2] Guy Goren and Alfonso de la Rocha. 2023. FRC-0051: Synchronous Consistent Block Broadcast for EC Security. https://github.com/filecoin-project/FIPs/blob/master/FRCs/frc-0051.md [Online; accessed 05-February-2024].

[3] Aggelos Kiayias, Saad Quader, and Alexander Russell. 2020. Consistency of Proof-of-Stake Blockchains with Concurrent Honest Slot Leaders. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. IEEE Computer Society, Los Alamitos, CA, USA, 776–786. https://doi.org/10.1109/ICDCS47774.2020.00065

[4] Xuechao Wang, Sarah Azouvi, and Marko Vukolić. 2023. Security Analysis of Filecoin's Expected Consensus in the Byzantine vs Honest Model. In *5th Conference on Advances in Financial Technologies (AFT 2023) (Leibniz International Proceedings in Informatics (LIPIcs))*, Joseph Bonneau and S. Matthew Weinberg (Eds.), Vol. 282. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 5:1–5:21. https://doi.org/10.4230/LIPIcs.AFT.2023.5

[5] Wikipedia contributors. 2023. Skellam distribution — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Skellam_distribution&oldid=1161277886 [Online; accessed 31-January-2024].