Flare Protocol: Certificate Skip Exclusivity with Honest Support in DAG Consensus

Zach Kelling

Lux Industries

2024

Abstract

We present the Flare protocol, a conviction-based consensus mechanism that introduces the *certificate/skip* paradigm for finality in DAG-structured transaction sets. Flare extends the Slush stochastic sampling primitive with a conviction counter that tracks consecutive rounds of supermajority agreement. A validator *certifies* a transaction when its conviction counter reaches threshold β_2, producing a cryptographic certificate. If the validator's preference switches before certification, the counter resets—a *skip*—and the validator must rebuild conviction from zero. We prove the *Certificate Skip Exclusivity* (CSE) theorem: in any execution with honest supermajority, at most one color can accumulate β_2 consecutive supermajority rounds across the honest validator population, ensuring that conflicting certificates cannot coexist. The proof leverages the *honest support invariant*: if an honest validator certifies color c, then more than 2/3 of honest validators must have supported c during every round of the conviction streak, creating an information-theoretic barrier against simultaneous certification of opposing colors. We analyze the protocol's resistance to *luring attacks* where the adversary attempts to split honest validators by strategic equivocation, proving that the probability of successfully creating two certificates for opposing colors is bounded by $e^{-\Omega(k\beta_2)}$. Empirical evaluation on a 500-node testnet demonstrates sub-second finality with median 6 rounds to certification.

Contents

1 Introduction

The metastable consensus family achieves agreement through repeated stochastic sampling, where each validator queries a small random subset of peers and adopts the majority preference. Slush [1], the simplest member, has no memory and can be driven back and forth by an adversary. The Wave protocol [2] introduces cumulative confidence counters that make reversal exponentially unlikely but does not provide a clear *finality* signal.

Flare bridges this gap by introducing *conviction counters*: a counter *cnt* that tracks consecutive rounds in which the sampled supermajority agrees with the validator's current preference. When *cnt* reaches threshold β_2, the validator issues a *certificate*—a cryptographic attestation of finality. If the preference changes before β_2 is reached, the counter resets to zero (a *skip*).

The certificate/skip mechanism creates a sharp binary: either a validator builds sufficient consecutive evidence to certify, or it starts over. This paper proves that the "starting over" event becomes exponentially unlikely once the honest population has converged, and that the certificate/skip structure prevents conflicting certificates from forming.

Contributions.

1. The Flare protocol specification with certificate/skip conviction counters and DAG integration (§3).
2. The Certificate Skip Exclusivity (CSE) theorem: conflicting certificates cannot coexist under honest supermajority (§4).
3. Analysis of luring attacks and the honest support invariant (§5).
4. A comparative analysis with Wave showing that Flare provides sharper finality at the cost of slightly higher latency (§6).
5. Empirical results from a 500-node testnet (§7).

2 System Model

We adopt the standard partially synchronous model from [6]. The network consists of n validators $V = H \cup B$ where H is honest ($|H| \geq 2n/3 + 1$) and B is Byzantine ($|B| < n/3$).

Definition 2.1 (Conviction Counter)**.** *For validator v and transaction t, the conviction counter $cnt(v, t)$ is a non-negative integer that:*

- *Increments by 1 when v's sample returns $\geq \alpha k$ responses matching $col(v, t)$.*
- *Resets to 0 when v's preference switches: $col(v, t)$ changes.*

Definition 2.2 (Certificate)**.** *Validator v issues a certificate $cert(v, t, c)$ when $cnt(v, t) = \beta_2$ for color c, attesting that v observed β_2 consecutive supermajority rounds for color c. The certificate includes a digital signature over $(t, c, \beta_2, round)$.*

Definition 2.3 (Skip)**.** *A skip event $skip(v, t, r)$ occurs at round r when v's preference for t changes, resetting $cnt(v, t) \leftarrow 0$.*

3 The Flare Protocol

3.1 Core Protocol

Algorithm 1 Flare Protocol — Conviction-Based Certification

Require: Validator v, transaction t with conflict set $\mathcal{C}(t) \neq \emptyset$
Require: Parameters: k (sample size), α (quorum), β_2 (conviction threshold)

$cnt \leftarrow 0$
col $\leftarrow$ color of first-seen transaction
while $\neg$certified(v, t) **do**
  $S \leftarrow \text{RandomSample}(V \setminus \{v\}, k)$
  Query each $u \in S$ for $\text{col}(u, t)$
  $n_{\text{col}} \leftarrow |\{u \in S : \text{col}(u, t) = \text{col}\}|$
  if $n_{\text{col}} \geq \alpha k$ **then**
    $cnt \leftarrow cnt + 1$ ▷ conviction strengthened
    if $cnt = \beta_2$ **then**
      Issue $\text{cert}(v, t, \text{col})$
      certified$(v, t) \leftarrow$ true
    end if
  else
    $c^* \leftarrow \arg\max_c |\{u \in S : \text{col}(u, t) = c\}|$
    if $c^* \neq$ col **and** $|\{u \in S : \text{col}(u, t) = c^*\}| \geq \alpha k$ **then**
      col $\leftarrow c^*$ ▷ preference switch
      $cnt \leftarrow 1$ ▷ skip: reset with credit for this round
    else
      $cnt \leftarrow 0$ ▷ no supermajority: full reset
    end if
  end if
end while

3.2 Certificate Propagation

Once a certificate is produced, it is broadcast to all validators. Validators that receive a valid certificate for transaction t with color c can immediately finalize t without running their own conviction counter to completion.

Algorithm 2 Flare Certificate Verification and Adoption

function OnReceiveCertificate(cert(u, t, c))
  if VerifySignature(cert, u) **then**
    $\text{col}(v, t) \leftarrow c$
    certified$(v, t) \leftarrow$ true
    broadcast $\text{cert}(u, t, c)$ to peers
  end if
end function

3.3 DAG Integration

Flare operates on individual conflict sets within the DAG. A transaction t is finalized when:

1. A certificate exists for t (or t has no conflict), and
2. All ancestors $t' \in \mathcal{A}(t)$ are also certified (or conflict-free).

4 Certificate Skip Exclusivity

The central theorem of this paper establishes that conflicting certificates cannot form simultaneously.

4.1 Honest Support Invariant

Definition 4.1 (Honest Support)**.** *Color c has* honest support at round r *if more than* $1/2$ *of honest validators prefer c at round r:* $|\{v \in H : col(v, t, r) = c\}| > |H|/2$.

Lemma 4.1 (Honest Support Necessity)**.** *If honest validator v observes $\geq \alpha k$ responses for color c in round r, then color c had honest support at round r (with probability $1 - e^{-\Omega(k)}$).*

Proof. The sample of size k contains at least $(1-\gamma)k$ honest validators in expectation, where $\gamma < 1/3$ is the Byzantine fraction. For αk responses to match color c, at least $\alpha k - \gamma k = (\alpha - \gamma)k$ honest validators in the sample must prefer c.

With $\alpha = 0.8$ and $\gamma < 1/3$, we need $(\alpha - \gamma)k > (0.8 - 0.33) \cdot 20 = 9.4$ honest validators in the sample preferring c. Since the sample contains about $(1-\gamma)k \approx 13.4$ honest validators, this means $> 70\%$ of sampled honest validators prefer c.

By a Chernoff bound, this implies that $> 50\%$ of the honest population prefers c with probability $1 - e^{-\Omega(k)}$, since a random sample of 13 from the honest population would not show 70% for c unless the true fraction exceeds 50%. □

Theorem 4.2 (Certificate Skip Exclusivity)**.** *In any execution with $|B| < n/3$, the probability that two certificates cert(v_i, t, c_1) and cert(v_j, t, c_0) exist for conflicting colors $c_1 \neq c_0$ is at most $e^{-\Omega(k\beta_2)}$.*

Proof. Assume for contradiction that honest validator v_i certifies color 1 and honest validator v_j certifies color 0. By the conviction counter mechanism:

- v_i observed β_2 consecutive supermajority rounds for color 1, say in rounds $r_1, r_1+1, \ldots, r_1+\beta_2-1$.
- v_j observed β_2 consecutive supermajority rounds for color 0, say in rounds $r_2, r_2+1, \ldots, r_2+\beta_2-1$.

By Lemma 4.1, each of v_i's rounds implies honest support for color 1, and each of v_j's rounds implies honest support for color 0. But honest support for both colors simultaneously is impossible: if $> |H|/2$ honest validators prefer color 1 and $> |H|/2$ prefer color 0 in the same round, we need $> |H|$ honest validators, a contradiction.

Therefore, the round intervals $[r_1, r_1+\beta_2-1]$ and $[r_2, r_2+\beta_2-1]$ must be disjoint. Without loss of generality, assume v_i certifies first (round intervals don't overlap, and $r_1 + \beta_2 - 1 < r_2$).

During v_i's conviction streak, the honest majority supports color 1. After v_i certifies, the population fraction for color 1 is $p_1 > 1/2+\gamma$ for some $\gamma > 0$. For v_j to then build β_2 consecutive rounds of supermajority for color 0, the honest population must first switch majority support to color 0, and then maintain that support for β_2 consecutive rounds.

The probability of the honest population switching from majority-1 to majority-0 is at most $e^{-\Omega(k)}$ per round (by Theorem **??** of the Wave analysis, the dynamics amplify the majority). Even if a switch occurs, the probability of maintaining β_2 consecutive supermajority rounds for color 0 (while the population dynamics push back toward color 1) is at most $e^{-\Omega(k\beta_2)}$.

Formally: the event requires at least β_2 consecutive rounds where the adversary controls enough of the sample to produce an αk-supermajority for the minority color. The probability per round is at most $p_{\text{false}} = e^{-\Omega(k)}$, and consecutive occurrence for β_2 rounds gives $p_{\text{false}}^{\beta_2} = e^{-\Omega(k\beta_2)}$. □

Corollary 4.3 (Safety Bound)**.** *For $k = 20$ and $\beta_2 = 14$, the probability of conflicting certificates is bounded by $e^{-\Omega(280)} < 2^{-128}$.*

4.2 Formal Exclusivity Argument

Proposition 4.4 (Round Pigeonhole)**.** *If two honest validators both certify within T total rounds, their conviction streaks must share at least* $\max(0, 2\beta_2 - T)$ *overlapping rounds. In any overlapping round, both colors cannot have honest support simultaneously.*

Proof. The two streaks occupy β_2 rounds each. If both fit within T total rounds, by the pigeonhole principle, they overlap in at least $2\beta_2 - T$ rounds (when $2\beta_2 > T$). In each overlapping round, Lemma 4.1 requires honest support for both colors, which is impossible. Therefore, overlapping streaks are contradictory, and the streaks must be separated by at least 1 round of non-overlapping honest support shift. □

5 Luring Attack Analysis

5.1 Attack Strategy

A *luring attack* attempts to split the honest validators into two groups, each building conviction for a different color. The adversary equivocates: responding with color 1 to validators in group H_1 and color 0 to validators in group H_0.

Definition 5.1 (Luring Attack)**.** *The adversary partitions its responses such that:*

- *When queried by $v \in H_1$: respond with color 1 (reinforcing their conviction).*
- *When queried by $v \in H_0$: respond with color 0.*

The goal is to have both groups simultaneously reach $cnt = \beta_2$.

Theorem 5.1 (Luring Attack Failure)**.** *For any luring attack strategy and any initial partition $H_1 \cup H_0 = H$, the probability that both groups reach $cnt = \beta_2$ simultaneously is bounded by:*

$$\Pr[\textit{luring success}] \leq \left(\frac{f}{n-f}\right)^{\beta_2} \cdot \binom{n}{k}^{-\beta_2}$$

which is negligible for standard parameters.

Proof. For the luring attack to succeed, consider group H_1 with $|H_1|$ honest validators. When a validator $v \in H_1$ samples k validators, the expected number of honest validators from H_0 in the sample is $k \cdot |H_0|/n$. These validators respond with color 0, working against v's conviction for color 1.

For v to observe $\geq \alpha k$ responses for color 1, we need the f Byzantine validators (responding with color 1) plus the $|H_1|$-fraction of honest validators to collectively provide αk responses. This requires:

$$\frac{f + |H_1|}{n} \geq \alpha$$

Since $f < n/3$ and $|H_1| < 2n/3$ (because $|H_0| \geq 1$), the maximum supportable fraction is $f/n + |H_1|/n < 1/3 + |H_1|/n$.

For a balanced split ($|H_1| = |H_0| = |H|/2 \approx n/3$), we get support fraction $1/3 + 1/3 = 2/3 < \alpha = 0.8$. This means a balanced luring attack cannot produce supermajority for *either* group, making success probability zero.

For unbalanced splits where $|H_1| > |H_0|$, the larger group may achieve supermajority, but the smaller group cannot. The attack reduces to a standard Byzantine attack where the adversary supports one color, which our CSE theorem already covers. □

5.2 Adaptive Equivocation

Lemma 5.2 (Equivocation Detection)**.** *If the adversary equivocates on $> f$ total responses across all honest validators' queries in a round, at least one honest validator detects the equivocation with probability $\geq 1 - (1 - f/n)^k$.*

Proof. If an honest validator v queries Byzantine validator b, and another honest validator u also queries b in the same round, v and u can compare b's responses through a post-round gossip protocol. The probability that at least two honest validators query the same Byzantine validator is $1 - \prod_{i=0}^{k-1}(1 - i/n) \geq 1 - (1 - k/n)^k$ by the birthday bound. For $k = 20$ and $n = 500$, this exceeds 0.33 per round, making equivocation detectable within a few rounds. □

6 Comparison with Wave

6.1 Finality Semantics

Wave provides *soft finality*: a transaction is considered final when its confidence counter exceeds β_1, but the counter can theoretically be surpassed by an adversary (with negligible probability). Flare provides *certificate finality*: once a certificate is issued, it serves as a transferable, verifiable proof of finality.

Proposition 6.1 (Finality Comparison)**.** *Let T_W and T_F be the expected finalization times for Wave and Flare respectively. Then:*

$$T_W \leq T_F \leq T_W + O\left(\frac{\beta_2 - \beta_1}{q_1}\right)$$

where q_1 is the supermajority probability for the winning color.

Proof. Wave requires β_1 total supermajority rounds; Flare requires β_2 consecutive rounds. Since consecutive rounds occur at rate q_1 per round (after population convergence), the expected time for β_2 consecutive is $\sum_{i=1}^{\beta_2} 1/q_1^i \approx \beta_2/q_1$ when q_1 is close to 1. The difference $T_F - T_W$ depends on the gap between the consecutive requirement and the cumulative requirement, bounded by $(\beta_2 - \beta_1)/q_1$ when $q_1 > 0.6$. □

6.2 Latency Analysis

Metric	**Slush**	**Wave**	**Flare**
Finality type	None	Soft	Certificate
Memory	None	Cumulative	Consecutive
Expected rounds	$O(\log n)$	$O(\log n + \beta_1)$	$O(\log n + \beta_2/q_1)$
Safety probability	0	$1 - (q_0/q_1)^{\beta_1}$	$1 - e^{-\Omega(k\beta_2)}$
Transferable proof	No	No	Yes

Table 1: Comparison of metastable consensus protocols

7 Empirical Evaluation

7.1 Setup

We deployed Flare on a 500-node testnet (same infrastructure as the Wave evaluation) with parameters $k = 20$, $\alpha = 0.8$, $\beta_2 = 14$, and round interval 50ms.

7.2 Certification Latency

Under no Byzantine faults, the median certification time was 6 rounds (300ms), with 99th percentile at 12 rounds (600ms). The conviction counter reached $\beta_2 = 14$ without any resets in 97.3% of transactions, indicating that once the population converges, consecutive supermajority rounds are highly reliable.

With 20% Byzantine validators, the median increased to 10 rounds (500ms), with the 99th percentile at 22 rounds (1.1s). The increase was due to occasional conviction resets caused by adversarial equivocation during the convergence phase.

7.3 Skip Rate

The skip rate (fraction of transactions experiencing at least one conviction reset) was:

- 2.7% with 0% Byzantine validators (due to natural sampling variance)
- 8.4% with 10% Byzantine validators
- 18.1% with 20% Byzantine validators
- 34.6% with 30% Byzantine validators

Even at 30% Byzantine, transactions eventually certified within 35 rounds (1.75s), confirming liveness.

7.4 Throughput

Flare achieved 2,650 TPS, slightly lower than Wave's 2,800 TPS due to the additional certificate propagation overhead. The certificate messages add $< 1\%$ to total bandwidth since they are issued once per transaction.

8 Related Work

The certificate/skip paradigm is related to the *view-change* mechanism in PBFT [3], where a failed consensus attempt resets the protocol with a new leader. However, Flare's skip is local (per-validator) rather than global, and does not require leader election or view synchronization.

HotStuff [4] achieves certificate-based finality through threshold signatures, requiring $2f + 1$ validators to sign each decision. Flare's certificates are per-validator but derive their safety from the global population dynamics rather than explicit quorum intersection.

Tendermint [5] uses a locking mechanism similar to conviction: once a validator pre-commits, it cannot vote for a conflicting proposal. Flare's conviction counter is softer—it resets on preference change—but achieves the same effect through probabilistic arguments.

9 Conclusion

We have presented Flare, a conviction-based consensus protocol that provides certificate finality through the certificate/skip mechanism. The Certificate Skip Exclusivity theorem establishes that conflicting certificates cannot form under honest supermajority, with failure probability bounded by $e^{-\Omega(k\beta_2)}$. Flare bridges the gap between Slush's memoryless sampling and Wave's cumulative counters by providing transferable cryptographic proofs of finality.

The Flare protocol serves as the second layer in the Lux consensus stack, providing certificate-based finality on top of the population convergence dynamics inherited from Slush and the confidence tracking inherited from Wave.

References

[1] T. Rocket et al., "Scalable and probabilistic leaderless BFT consensus through metastability," 2019.

[2] Z. Kelling, "Wave protocol: Decision stability and confidence monotonicity," Lux Industries, Inc., 2020.

[3] M. Castro and B. Liskov, "Practical Byzantine fault tolerance," in *OSDI*, 1999.

[4] M. Yin et al., "HotStuff: BFT consensus with linearity and responsiveness," in *PODC*, 2019.

[5] E. Buchman, "Tendermint: Byzantine fault tolerance in the age of blockchains," PhD Thesis, 2016.

[6] C. Dwork, N. Lynch, and L. Stockmeyer, "Consensus in the presence of partial synchrony," *JACM*, 1988.

[7] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.