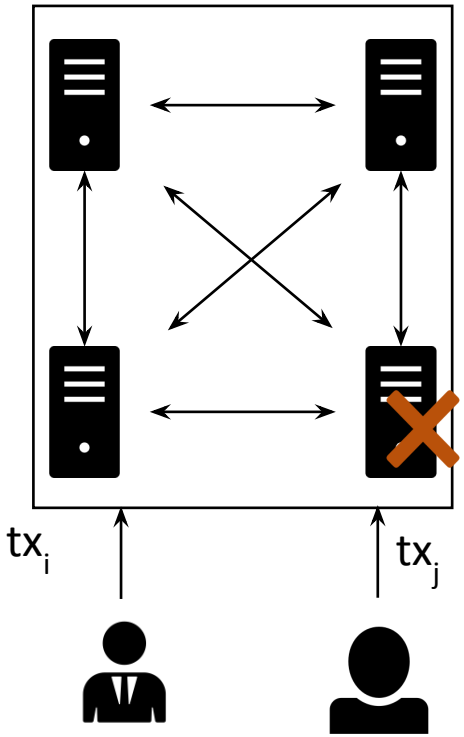
Hydrangea: Optimistic Two Round Partial Synchrony with Improved Fault Resilience

Nibesh Shrestha, Aniket Kate and Kartik Nayak

Usenix Security Symposium 2026

Byzantine Fault-Tolerant State Machine Replication



Safety: No two non-faulty servers commit on a different sequence of transactions

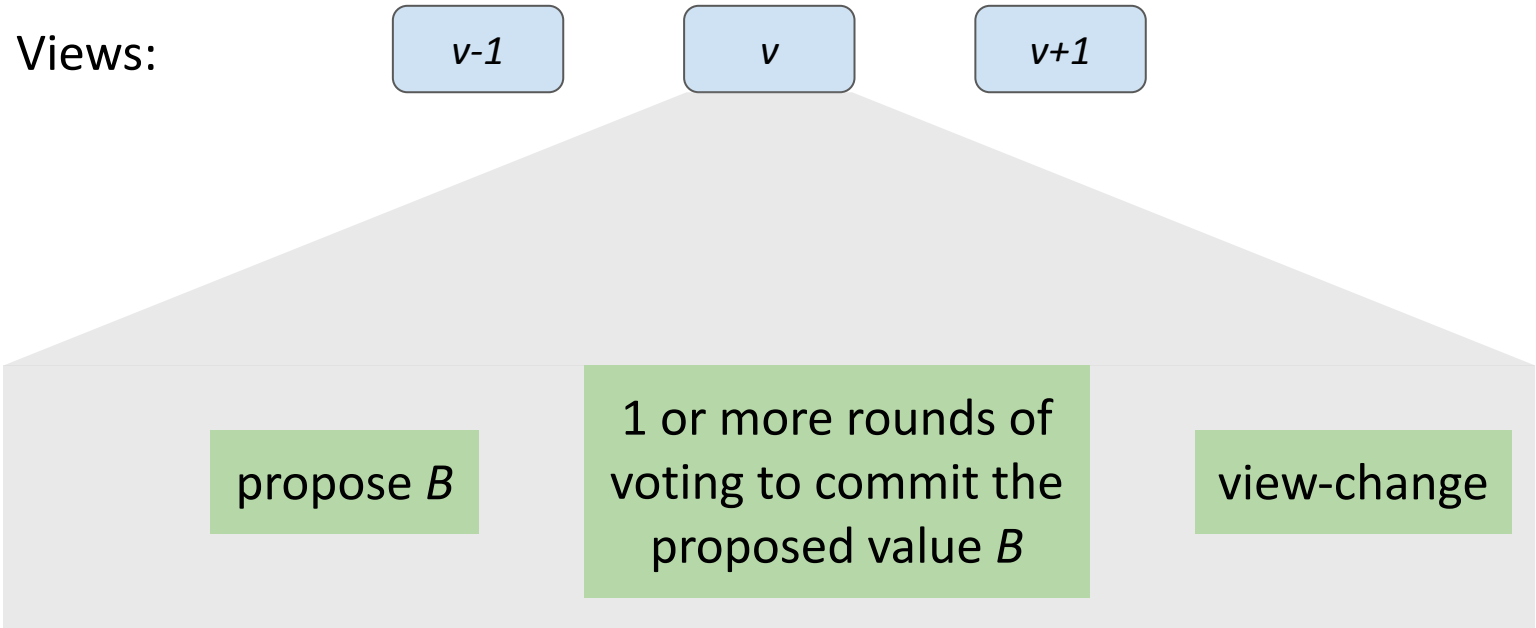
Liveness: Transactions issued by clients must eventually be committed by non-faulty servers

Partial Synchrony

- Messages may be arbitrarily delayed before GST
- After GST, messages are delivered within Δ time

GST: Global Stabilization Time

General Structure of BFT SMR Protocols



The Latency-Resilience Tradeoff

Setting	Resilience	Commit Latency
n=3f+1 PBFT'99, ..., Moonshot'24	*33%*	*3* rounds
n=5f+1 Minimmit'25	*20%*	*2* rounds
n=3f+2p+1 FaB'05, SBFT'19, Kudzu'25	*20% <= f < 33%* e.g. *25%*	*2* rounds, when *p < f* faults; *3* rounds otherwise

n: No. of parties, *f*: Byzantine fault tolerance, *p*: Fast path fault tolerance

Can we support optimistic two-round latency while still achieving high resilience?

A More Practical Byzantine-and-Crash Model

- ➢ Better reflects real-world failures
- ➢ Mostly failures are crashes, Byzantine faults are rare

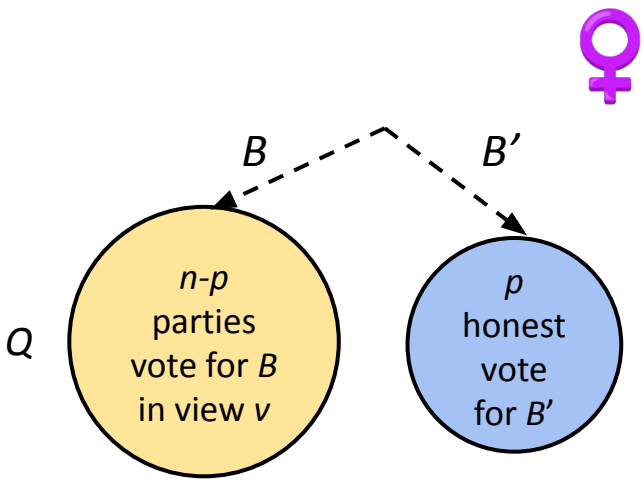
Solana's **Alpenglow**:

Under $n=3f+2c+1$ *with* $f = c \simeq 20\%$

- 3 round commit under *f* Byzantine *+ c* crash faults i.e., *40%* faults
- 2 round commit under $p = 20\%$ faults (Byzantine or crash)

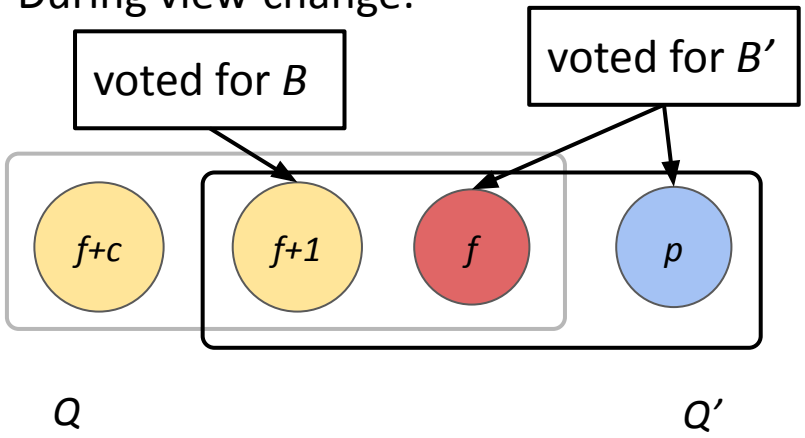
Alpenglow under Partial Synchrony

Under $n = 3f+2c+1$ with $f = c = p \simeq 20\%$



Some honest party commit B upon receiving $n-p$ votes for B

During view-change:



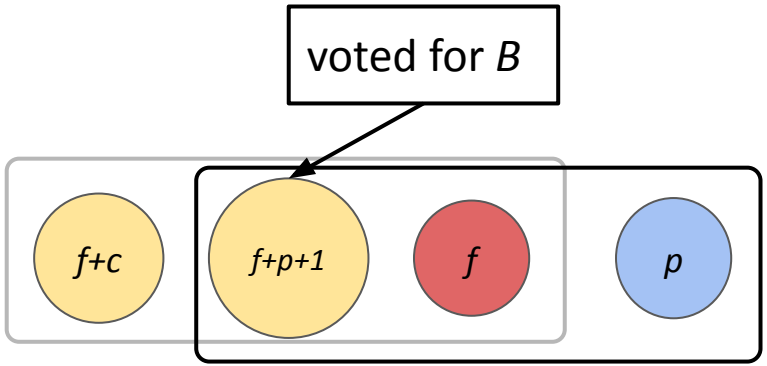
$Q \cap Q'$ contains at least $f+1$ honest parties who voted for B in view v. Remaining $f+p$ parties (including f Byz) in Q' voted B' in view v

Assumes no (or detectable) leader equivocation (requires synchrony)

Can we achieve optimistic two-round commits under Byzantine and crash faults while remaining safe under partial synchrony?

Hydrangea: Stronger Quorum Intersection Restores Safety

Under $n = 3f+2c+1$



$Q \cap Q'$ contains at least $f+p+1$ honest parties who voted for B in view v. Remaining $f+p$ parties (including f Byz) in Q' voted B' in view v

$n=3f+c+2p+1$

OR

For $n=3f+2c+1$:

$p = \lfloor c/2 \rfloor$

Safe under partial synchrony

When $f = c \simeq 20\%$, $p \simeq 10\%$

Hydrangea: The General Setting

Under $n = 3f+2c+k+1,\ k \geq 0$

- *2* rounds under $p = \lfloor (c+k)/2 \rfloor$ *faults*
- *3* rounds under *f* Byzantine *and c* crash faults

An example:

- $f=20, c=20$ *then* $p=10$
- $f=20, c=20$ *and* $k=20$ then $p=20$

n: No. of parties, *f*: Byzantine fault tolerance, *c*: crash fault tolerance, *p*: Fast path fault tolerance

Hydrangea as a Unifying Framework

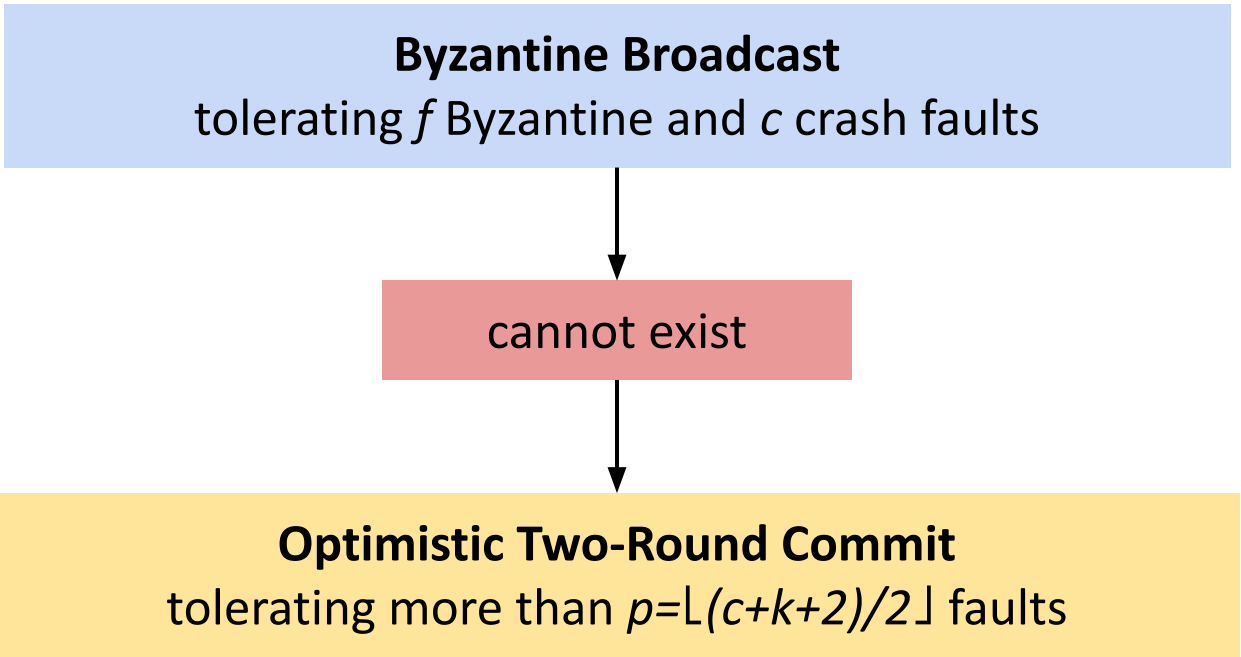
Under $n = 3f+2c+k+1,\ k \geq 0$

c	***k***	***p***	
0	*0*	*0*	Simplex'23
0	*2p*	*p*	Kudzu'25
0	*2f*	*f*	Minimmit'25
c	*2f+c*	*f+c*	Orcaella'26

n: No. of parties, *f*: Byzantine fault tolerance, c: crash fault tolerance, *p*: Fast path fault tolerance

Lower bound on Resilience for Optimistic Two-Round

Under $n=3f+2c+k+1$



The Hydrangea Protocol (Simplified)

Under $n = 3f+2c+1$ with $p = \lfloor c/2 \rfloor$

Propose. The leader L_v send-all $\langle propose, B, v, PC_{v-1} \rangle$

Vote. Upon receiving $\langle propose, B, v, PC_{v-1} \rangle$ from L_v, send-all $\langle vote, B, v \rangle$

Precommit. Upon receiving $2f+c+1$ votes for B (*i.e.*, $C_v(B)$), send-all $\langle commit, B, v \rangle$

Commit.

- **Fast path.** Upon receiving $n-p$ votes for B
- **Slow path.** Upon receiving $2f+c+1$ commit messages for B

$C_v(B)$: Certificate for B in view v PC_{v-1}: Progress certificate for view $v-1$

The Hydrangea Protocol Under General Setting

Under $n = 3f+2c+k+1$ with $p = \lfloor (c+k)/2 \rfloor$

Propose. The leader L_v send-all $\langle propose, B, v, PC_{v-1} \rangle$

Vote. Upon receiving $\langle propose, B, v, PC_{v-1} \rangle$ from L_v, send-all $\langle vote, B, v \rangle$

Precommit. Upon receiving ~~$2f+c+1$~~ $2f+c+\lceil k/2 \rceil+1$ votes for B (*i.e.*, $C_v(B)$), send-all $\langle commit, B, v \rangle$

Commit.

- **Fast path.** Upon receiving $n-p$ votes for B
- **Slow path.** Upon receiving $2f+c+1$ commit messages for B

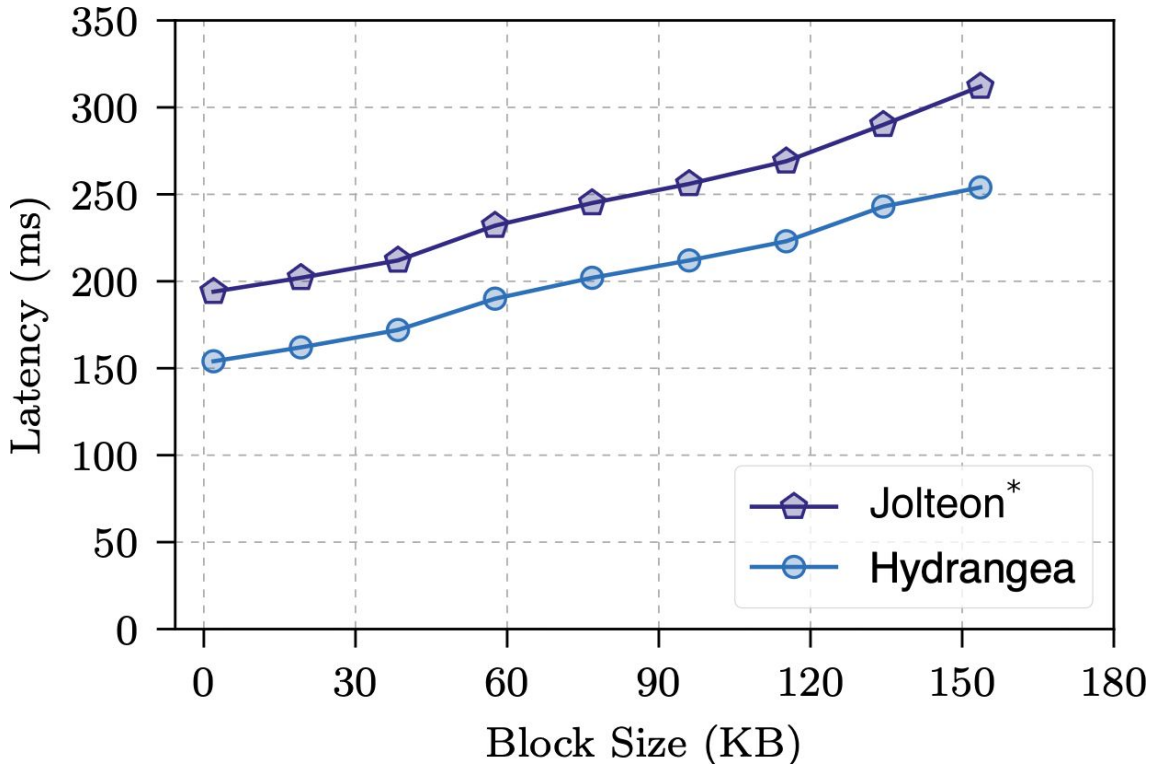
$C_v(B)$: Certificate for B in view v PC_{v-1}: Progress certificate for view $v-1$

Experimental Evaluation

For *n=50:*

Jolteon*: *n=3f'+1, with f'=33%* ;

Hydrangea: *n=3f+2c+1,*
with f=c≃20% and p ≃ 10%



Summary

Hydrangea breaks the traditional tradeoff between latency and resilience

- Achieves **optimistic two round commits** with improved fault resilience
- **Three round commits** under full adversarial conditions

Simultaneously achieve **low latency, strong resilience and robust performance** under a generalized fault model.

Hydrangea++[**S**K'25]

- Reduces block proposal time to one-round

Thank You!





Contact: nibeshrestha2@gmail.com