Lecture 7:
The Tendermint Protocol

(Lecture Series on Foundations of Blockchains)

Tim Roughgarden

Consensus with Partial Synchrony

Recall: Partially synchronous model. (unknown transition time GST from asynchrony to synchrony)

Goals: (for SMR) consistency (always) and liveness (eventually).

Lecture 6: if $f \geq n/3$, no protocol satisfies these goals (even with PKI).

Tendermint: an SMR protocol that, when $f < n/3$, satisfies consistency (always) + liveness (eventually). [Buchman/Kwon/Milosevic 2018]

Tendermint: High-Level Ideas

Idea#1: iterated single-shot consensus. (output of each = ordered list of txs ("block"))
⇒ each node i maintains its own "height" h_i
(in asynchronous phase, might be different for different nodes)
+ ignores all messages about other heights

Idea#2: for a fixed height, keep proposing + voting til agreement.

Idea#3: two stages of voting. (Necessary because different nodes may see different voting outcomes, need an intermediate "hedging" outcome.

Quorum Certificates (QCs)

Preliminaries: - assume PKI, all messages signed by sender

- round = interval of 4Δ timesteps (recall shared global clock, Δ known a priori)
  (all nodes agree on the current round, even in the asynchronous phase)
- use rotating leaders (one per round) [easy due to shared clock, permissioned setting]

Definition: a quorum certificate (QC) = batch of $\geq \frac{2}{3}n$ signed votes for some block B (at some height, at some round, at some stage). consequence: all QCs for same block # + round + stage must agree on the block (honest nodes don't double-vote)

Lemma: any two QCs overlap in at least one honest node.

Proof: overlap $\geq n - \frac{1}{3}n - \frac{1}{3}n = \boxed{\frac{1}{3}n > f}$. ← by assumption!

Plan: each node i maintains a (B_i, QC_i) pair. ← QC for the block B_i
→ initially QC_i = null, B_i = all unexecuted txs that i knows about

- periodically updates to most recent block-QC pair it's heard about
  (also save for future use any QCs for future blocks)

The Tendermint Protocol

Fix a height (e.g., block #9), a round r with leader ℓ.

$t = 4\Delta r$: ℓ updates (B_ℓ, QC_ℓ) to most recent QC known, proposes to all other nodes.

$t = 4\Delta r + \Delta$: if node i receives (B_ℓ, QC_ℓ) from ℓ, with QC_ℓ at least as recent as QC_i:

- broadcast first-stage vote for B_ℓ
- update $(B_i, QC_i) := (B_\ell, QC_\ell)$
- broadcast (B_ℓ, QC_ℓ) to all other nodes

$t = 4\Delta r + 4\Delta$: (just before 1st phase of round $r+1$)

if know a stage-2 QC for block $\#h_i$ supporting a block B, commit B to local history, increment h_i. (repeat as needed)

$t = 4\Delta r + 2\Delta$: if node i receives $\geq \frac{2}{3}n$ round-r stage-1 votes for B:

- set $QC_i :=$ this QC, $B_i := B$
- broadcast second-stage vote for B_i
- broadcast (B_i, QC_i) to all other nodes

$t = 4\Delta r + 3\Delta$: if node i receives $\geq \frac{2}{3}n$ round-r stage-2 votes for B:

- set $QC_i :=$ this QC, $B_i := B$
- commit B to local history (!)
- broadcast (B_i, QC_i) to all other nodes
- increment h_i, re-initialize B_i & QC_i

In the background (at all times): store all QCs received for blocks $h_i+1, h_i+2, \ldots$

Tendermint: Proof of Consistency

Theorem: Tendermint satisfies SMR consistency. (for a given block #, all honest nodes commit the same block)

Proof: Fix a height h (e.g., block # 9).

Let r = 1st round in which $> n/3$ honest nodes [call this set S] cast stage-2 votes [prerequisite for creation of a stage-2 QC] for same block B^*.

Intuition: nodes of S "lock in" on B^*. Because $|S| > n/3$, can't create any new QCs for blocks other than B^*.

Formally: (by induction) at end of round r:
(i) $B_i = B^*$ for all $i \in S$
(ii) QC_i from round-r stage-1 or later
(iii) all QCs for other blocks are from round r or earlier

- in round $r+1$, no nodes of S change their mind => (i)-(iii) still hold at end of round

=> history repeats in rounds $r+2, r+3, \ldots$ (no QCs for blocks $B \neq B^*$, ever again)
=> can never have stage-2 QCs that support different blocks
=> will never have different honest nodes commit to different blocks at height h

QED!

Tendermint: Proof of Liveness

Claim: Tendermint satisfies SMR liveness (eventually).

Proof: Consider a transaction T known to all honest nodes.

=> fast forward to pair r_1, r_2 of consecutive rounds after GST + Δ with honest leaders l_1, l_2

must exist ($f < n/3$)

Lemma: at start of round r_1, every honest node working on block #h or h+1

Definition: a round is clean if (i) post-GST (ii) honest leader (iii) all honest nodes working on same block # (iv) after update in 1st phase, leader's QC at least as recent as that at any honest node.

Lemma: clean round => all honest nodes commit the block proposed by leader.

Case 1: all honest nodes start round r_1 working on block #h+1. (round r_1 is clean, done)

Case 2: all working on block #h. Round r_1 clean, commits some block. Round r_2 clean, commits block including T.

Case 3: leader but not all honest nodes working on block #h.

=> round r_2 clean with honest block proposal, done.

Case 4: leader but not all honest nodes working on block h+1.

Can We Do Better?

Main Result: if $f < n/3$, Tendermint satisfies consistency (always) and liveness (eventually) in the partially synchronous model.

Possible improvements?:

- can't increase # of Byzantine nodes (without compromising elsewhere)
- can't relax partially synchronous to asynchronous (" " ")
- can't strengthen "eventually" to "always" (" " ")
- alternative trade-offs (e.g. favor liveness over safety during an attack)
  - as in longest-chain consensus (Bitcoin, Ethereum, etc.)
- same guarantees (fault-tolerance, consistency, eventual liveness) but better performance (smaller communication complexity, fewer rounds, faster recovery time post-GST, etc.)
  - see HotStuff (Facebook Diem), Casper-FFG (Ethereum)