L04 — Consensus Mechanisms

Cheatsheet

Byzantine Generals Problem

Core Result (Lamport et al. 1982)

Need $n \geq 3f + 1$ nodes to tolerate f traitors.
Equivalently: honest nodes must exceed $\frac{2}{3}$ of total.

Setup: n generals agree on attack/retreat over unreliable channels; up to f may send conflicting (Byzantine) messages.

Why 3f+1? Need $2f+1$ honest for majority; f extra absorb conflicting traitor messages. Consensus impossible with $n \leq 3f$.

Blockchain link: Bitcoin solves BGP in open networks via PoW — economic cost replaces the requirement for a known participant list (counters Sybil attacks).

BFT (Byzantine Fault Tolerant) — system property: consensus reached despite up to f Byzantine (arbitrary/malicious) failures.

Proof of Work (PoW)

Mining Puzzle

Find nonce n s.t. $H(\text{block header} \| n) < \text{target}$.
Difficulty $\propto 1/\text{target}$. Verification: one hash.

Mining process:

1. Collect transactions from mempool
2. Build block header with Merkle root
3. Iterate nonce; compute SHA-256 hash
4. Broadcast if hash < target; earn reward + fees

Difficulty adjustment (Bitcoin: every 2016 blocks, ≈2 weeks):

$$\text{New Difficulty} = \text{Old Difficulty} \times \frac{\text{Actual time}}{\text{Target time (2 weeks)}}$$

51% attack: majority hash power can rewrite recent blocks and double-spend own transactions; cannot steal others' funds or create coins out of thin air. Economically irrational for Bitcoin.

Block reward (Bitcoin): 50 BTC genesis → halves every 210,000 blocks. Post-4th halving (April 2024): 3.125 BTC. Final BTC ≈ 2140.

Proof of Stake (PoS)

Core Idea

Replace computational work with locked economic stake.
Validators selected (weighted-randomly) by staked amount.
Security = capital at risk, not energy expended.

Validator selection: weighted-random per slot; Ethereum uses 12-second slots with minimum 32 ETH stake.

Attestations: committees of validators vote on each block; correct attestations earn rewards, absence incurs minor leaking penalty.

Nothing-at-stake problem: in naive PoS, validators can vote on multiple forks for free. Solution: **slashing**.

Slashing conditions:

- Double signing (proposing two conflicting blocks)
- Surround voting (conflicting attestations)
- Penalty: fraction up to 100% of staked ETH; forced exit

Long-range attacks: attacker rewrites history using old keys. Mitigated by *weak subjectivity* (trust recent checkpoint) and social consensus.

PoW vs. PoS Comparison

Property	PoW	PoS
Resource used	Electricity (ASICs)	Locked capital
Energy	Very high	Low (−99.95% ETH Merge)
Finality	Probabilistic	Econ. final (≈13 min ETH)
Sybil resistance	Hash rate	Stake
Attack cost	51% hash power	33% stake
Centralization	ASIC pools	Liquid staking
Block time	≈10 min (BTC)	12 s (ETH)
TPS (approx.)	3–7	15–30
Permissionless	Yes	Yes (32 ETH min)
Example	Bitcoin	Ethereum

Delegated Proof of Stake (DPoS)

Token holders **vote** for a fixed set of delegates (block producers). Delegates can be voted out at any time.

Examples: EOS (21 block producers), Tron, Lisk.

Trade-offs:

+ High throughput, fast finality (1000+ TPS)
+ Low latency; efficient block production
- More centralised (small fixed delegate set)
- Potential for delegate collusion

DPoS sacrifices decentralisation for performance.

BFT Variants

Protocol	Used In	Key Properties
PBFT	Hyperledger Fabric	3-phase voting; deterministic finality; $O(n^2)$ msgs; known validator set
Tendermint	Cosmos, Polygon	PBFT-inspired; instant finality; light-client friendly
HotStuff	Diem, Solana	Linear msgs ($O(n)$); pipelined; leader-based

All BFT variants: tolerate up to $\lfloor (n-1)/3 \rfloor$ faulty nodes; require **known validator set** (permissioned); do not scale well beyond ∼100 validators due to communication overhead.

Finality: BFT mechanisms provide **deterministic finality** — once a block is committed it cannot be reverted.

Selfish Mining (Eyal & Sirer 2013)

Strategy: Mine blocks but withhold from broadcast; build a secret longer chain; release selectively to orphan honest blocks.

Why it works: Honest miners waste work on blocks that get orphaned; attacker earns > proportional reward share.

Profitability threshold: effective with as little as **25–33%** of hash rate (exact threshold depends on network connectivity parameter γ).

Selfish Mining Revenue Condition

Revenue ratio $> \alpha$ when:
$\alpha > \frac{1-\gamma}{3-2\gamma+\gamma}$ where $\gamma \in [0,1]$ is attacker's propagation advantage.

Defences: uniform tie-breaking, shorter block times, monitoring.

MEV — Maximal Extractable Value

Definition

Value extractable by controlling transaction ordering within a block.
Formerly "Miner Extractable Value"; now applies to PoS validators/builders.

MEV strategies:

- **Front-running**: Copy pending profitable TX, submit with higher gas
- **Sandwich attack**: Buy before victim TX, sell after (DEX price manipulation)
- **Back-running**: Arbitrage immediately after a large trade
- **Liquidations**: Race to liquidate undercollateralised positions

Impact on users: worse prices, failed transactions, wasted gas.

Ethereum solutions:

- **Flashbots**: Private mempool; reduces gas wars
- **MEV-Boost**: Separates block *builder* from block *proposer*
- Intent-based systems (e.g. CoW Protocol): hide TX details

MEV cannot be eliminated, only redistributed.

Finality Types

Type	Description
Probabilistic	PoW: probability of revert decreases with confirmations. Bitcoin: 6 blocks ≈60 min considered final
Economic	PoS with slashing: revert is theoretically possible but prohibitively costly (destroys attacker stake)
Deterministic	BFT protocols: once committed, absolutely final; no revert possible under $< f$ failures

Ethereum PoS: economic finality achieved after 2 epochs (≈13 min); slashing ensures attack would cost >33% of total staked ETH to reverse.

Key Parameters

Network	Block Time	Mechanism	Reward / Yield
Bitcoin	≈10 min	PoW (SHA-256)	3.125 BTC
Ethereum	12 s	PoS (Gasper)	≈3–5% APY
Solana	0.4 s	PoH + PoS	≈6–7% APY
Cosmos	≈6 s	Tendermint	≈15% nominal
EOS	0.5 s	DPoS	≈1–2% APY