Nakamoto Consensus

Benedikt Bünz
Consensus

• Security Properties:
  • Consistency: Honest nodes do not contradict
  • Liveness: Progress is made

• Network Models
  • Synchronous: Messages get delivered immediately
  • Partially Synchronous: Messages are out of order
Consensus Committee Leader Accepts/Rejects TXs Committee Leader
Problems with approach

- Known committee
  - (must communicate)
- Large committee
  - Large communication
- Honest majority (incentives)
- Predictable Leader
  - Bribing 💰
Recap

genesis block

BH₁

version (4 bytes)
prev (32 bytes)
time (4 bytes)
bits (4 bytes)
nonce (4 bytes)
Tx root (32 bytes)

80 bytes

BH₂

prev

Tx root

BH₃

prev

Tx root

...
Nakamoto Consensus

PoW: Find nonce s.t. $H(\text{Block}) < \text{Target}$

$\text{Target s.t. blocks found every 10 min}$
Nakamoto Consensus

```
<table>
<thead>
<tr>
<th>H</th>
<th>HH</th>
<th>H</th>
<th>HH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Genesis</td>
<td>BH₁</td>
<td>BH₂</td>
<td>BH₃</td>
</tr>
<tr>
<td>Prev Time Nonce Root</td>
<td>Prev Time Nonce Root</td>
<td>H(BH₂) Time Root</td>
<td>H(BH₂) Time Nonce Root</td>
</tr>
<tr>
<td>coinbase Tx</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Nakamoto Consensus

- Miners “race” to add blocks
  - Need to find PoW solution
  - Probability winning ~ Computation power
  - One winner every ~10 min
  - Target adjusted every 2 weeks
- Leader election/race combined with tx adding
- (Honest) miners extend longest chain
- Timestamps must be roughly accurate
- All transactions must be valid
- Blocks/Transactions become final after $k$ blocks

PoW:
Find nonce s.t.
$H(\text{Block}) < \text{Target}$
Forks and Orphans

- Working on B
- Working on A
Forks and Orphans

Orphaned block

Working on B C

Working on A C
Preventing double spends

I’ll just produce a different chain

I’ll wait k blocks

Here are the keys

No Car TX

We’ll be working on the longest chain

3 BTC For Car
51% Attack

I’ll just produce a different chain

New longest chain
1. **Consistency.** Honest nodes agree on all but last $k$ blocks (except with prob. $O(2^{-k})$)

2. **Chain quality.** Any consecutive $k$ blocks contain “sufficiently many” honest blocks (except with prob. $O(2^{-k})$). **Miners controlling $p$ fraction of power should roughly mine $p$ fraction of blocks.**

3. **Chain growth.** Chain grows at a steady rate.

   *$g$-chain growth: Growth by $k$ blocks every $k/g$ “rounds”*
Nakamoto properties => Blockchain

• Consistency implies Blockchain consistency

• Chain growth + chain quality implies Blockchain liveness
  - The chain grows by k blocks every k/g periods
  - By chain quality, a high fraction of blocks are contributed by honest miners, and therefore include all transactions they heard so far
Nakamoto consensus

**Consistency intuition:** Suppose adversary has 49% power

- Adversary can fork chain by 1 block faster than honest miners extend current chain w/ prob. close to \( \frac{1}{2} \), or by 2 with prob. \( \frac{1}{4} \)
  - No problem! If adversary broadcasts fork, everyone switches, this is now the longest chain

- What if miner forks chain 6 blocks deep and doesn’t broadcast until it has a longer chain than honest?
  - Probability \( \frac{1}{64} \) it mines 6 blocks before honest mines 1
  - Probability \( < 8 \times 2^{-7} \) it mines 7 blocks before honest mines 2
  - What is probability adversary ever catches up?
Consistency intuition: (continued…)

Suppose adversary has $p < 1/2$ fraction of power. What is the probability adversary catches up from 6 blocks behind?

- *Simplified model*: repeated rounds, in every round adversary catches up by 1 block with probability $p$, and falls behind by 1 block with probability $1 - p$.
- Biased random walk on number line starting at 0, +1 with probability $p$ and -1 with probability $1 - p$. Probability walk ever reaches 6?
- Probability $P_z$ that walk ever reaches $+z$ is $\left(\frac{p}{1-p}\right)^z$ (e.g. $p = 1/3$, then $P_6 < 0.0062$)
Nakamoto consensus

What goes wrong if adversary has $p > 1/2$ power?

• Adversary’s private fork grows at faster rate than honest chain

• For any $k$, adversary starts $k$ blocks behind, will eventually catch up to length of honest chain
45% Attack

I’ll just produce a different chain

Incur network delays and orphans

Cloud

No Car TX

3 BTC For Car
Nakamoto consensus

Network delay & work difficulty

• What happens if miners can solve puzzles faster than they can propagate solutions through network?
• Adversary might receive the next valid block $\Delta$ steps ahead of the other honest nodes ($\Delta = \text{delay}$)

$\Rightarrow$ Adversary starts working on next puzzle with a $\Delta$ time head start over other honest nodes

$O(\Delta)$ “free” hash trials
Nakamoto consensus

Adjusting difficulty for $\Delta$

\[ \alpha (1 - \alpha \Delta) > \beta \]

Intuition:
If `block-time' is $c\Delta = \frac{1}{\alpha}$ (i.e. honest puzzle solved every $c\Delta$ steps)
Then on average, honest nodes waste $\Delta$ steps of work every $c\Delta$ steps, while adversary never wastes work. So “effective” reduced honest rate is

Formula from [PSS '16] building on [GKL15, SZ15]
Nakamoto consensus

Adjusting difficulty for $\Delta$

$$\alpha(1 - 2\alpha(\Delta + 1)) > \beta$$

Intuition:
If `block-time' is $c\Delta = \frac{1}{\alpha}$ (i.e. honest puzzle solved every $c\Delta$ steps)
Then on average, honest nodes waste $\Delta$ steps of work every $c\Delta$ steps, while adversary never wastes work. So “effective” reduced honest rate is
$$\alpha \left( \frac{c}{c+1} \right) \approx \alpha \left( \frac{c-1}{c} \right) = \alpha \left( 1 - \frac{1}{c} \right) = \alpha (1 - \alpha \Delta)$$

Formula from [PSS ‘16] building on [GKL15, SZ15]
Red line = min $p$ value for which attack from PSS works.

Nakamoto magically chose $c = 60$ (10 min blocktime assuming 10s network delay)

Blue line = max value of $p$ s.t. $\frac{\beta}{\alpha} = \frac{p}{1-p}$ and $\frac{\beta}{\alpha} < 1 - 2(\Delta + 1)\alpha$
Short Forks and Liveness

Long forks are impossible but short forks may not be. This is a liveness issue.

Need to ensure that some “honest” blocks are in the longest chain.

Could be used to censor transactions.
Nakamoto chain quality

- Chain Quality is percentage of honestly mined blocks
  - Honest mined blocks include all transactions!
  - Prevents censorship
- Say the adversary controls a $p$ fraction of the mining power $p < \frac{1}{2}$
- Ideally honest parties mine a $1 - p$ fraction
- Can prove they mine at least $1 - \frac{p}{1-p}$
  
  \[
p = \frac{1}{3} \rightarrow Q = \frac{1}{2}
  \]

If $p > \frac{1}{2}$ then adversary could mine every block in worst case

\[\Rightarrow \text{chain quality is 0}\]
• For every $p < \frac{1}{2}$, if mining difficulty is appropriately set as function of network delay $\Delta$ then Nakamoto consensus guarantees:

1. Consistency (for $\alpha, \beta, \Delta$ satisfying formula)

2. Chain quality: $1 - \frac{p}{1-p}$ fraction blocks honest

3. $O(1/\Delta)$-Chain growth
Nakamoto Consensus and Partial Synchrony

- Nakamoto Consensus can be secure up to $\frac{1}{2}$ corruptions
- Can tolerate network delays
- Contradicts partial synchrony lower bound?
  - No
  - Protocol needs a bound on delays (c)
  - Consistency broken even with honest nodes
Nakamoto Properties

- Anonymous participation
- Nodes can join/leave
  - Very scalable
  - Sleeping Beauty property
- Leader not known beforehand
  - Makes bribing harder
- Up to ½ corruptions

- Slow
  - Even when everyone is honest
- Resource intensive
  - PoS based possible
- No finality
- No guarantees under long delays
Incentives

- Mining (solving PoW puzzles) is very expensive
- *Honest* majority does not seem realistic
- Satoshi’s genius idea: Combine issuance and rewards
- Block reward only paid if block part of longest chain
- High Variance -> Mining Pools
Incentives

Large opportunity cost for unsuccessful attacks
Selfish mining attack

Attacker has 1/3 of mining power. Miner is rational (maximize rewards)

Keeps block private

Once attacker has a two block lead he can mine until honest chains catch up
Selfish mining attack

Attacker has 1/3 of mining power. Miner is rational (maximize rewards)

Keeps block private

Once attacker has a two block lead he can mine until honest chains catch up

Attacker publishes chain and invalidates honest blocks
Selfish mining attack

Attacker has 1/3 of mining power. Miner is rational (maximize rewards)

Keeps block private

If honest miners finds block: Publish and it’s a block race (Attacker has at least 1/3 p of winning)
Selfish mining analysis

Honest reward = 1

P Block Race: 2/3

Win: 1/3 chance
2 of 3 blocks
Reward 2

Loose: 2/3 chance
Reward 0

P Run away: 1/3

\[
\frac{2}{3} \times \frac{1}{3} \times 2 + \frac{1}{3} \times 2 = \frac{10}{9} > 1
\]

Reward > 2
Selfish Mining

Optimal Selfish mining

Explains why chain quality < 1 - p
No Attacks in Practice?

• Attacks possible but not seen
• Ghash.IO had >50%
  • Gave up mining power
  • No Selfish mining attacks
• Why?
  • Miners care about Bitcoin price
  • Not rational in $ terms to attack
  • Not guaranteed in the future
Next lecture:
Randomness beacons, VDFs, large scale PoS