EE 374: Fundamentals of Blockchain Infrastructure **Stanford, Spring 2025**

Lecture 15: Scaling Bitcoin

May 19, 2025

Lecturer: Robin Linus Scribe: Alex Wang

1 Introduction to Bitcoin and the Scaling Challenge

Bitcoin was introduced on October 31, 2008, by the pseudonymous Satoshi Nakamoto in the whitepaper "Bitcoin: A Peer-to-Peer Electronic Cash System." It proposed a decentralized system for online payments, eliminating the need for financial institutions by using a peer-to-peer network, digital signatures, and a consensus protocol to solve the double-spending problem.

From its inception, concerns about Bitcoin's ability to scale to a global transaction volume were raised. James A. Donald, on November 2, 2008, noted, "We very, very much need such a system, but the way I understand your proposal, it does not seem to scale to the required size." This highlighted the fundamental challenge: how to increase Bitcoin's transaction throughput (currently around 7 transactions per second) without compromising its core principles of decentralization and security.

Satoshi Nakamoto initially envisioned a system capable of handling large transaction volumes, suggesting users might not need to run full nodes but could use light clients or pruned nodes. However, a de facto block size limit of 1MB was introduced by Satoshi in July 2010, leading to significant debate as transaction volume grew and blocks began to fill up around 2015.

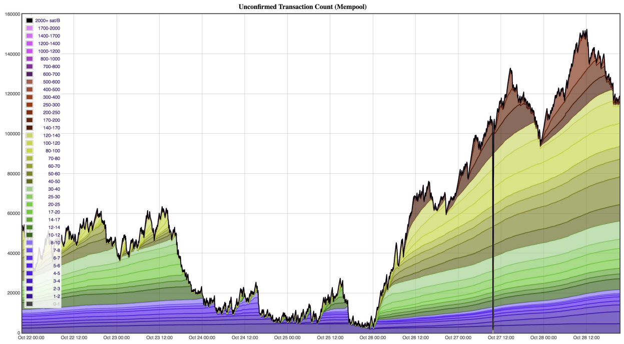

Figure 1: Bitcoin MemPool Volume

2 On-Chain Scaling: The "Big Blocks" Debate

The most direct approach to increasing transaction throughput on-chain is to increase the maximum block size. This was a central topic in the "Block Size War."

2.1 Arguments for "Big Blocks"

- **Trivial Code Change:** Modifying the block size parameter is relatively simple.

- **Great User Experience (UX):** Larger blocks could initially lead to faster confirmation times and lower fees by accommodating more transactions.
- **Profitable for Miners:** More transactions per block could increase fee revenue for miners.

2.2 Drawbacks of "Big Blocks"

- **Centralization:** Larger blocks increase bandwidth, processing, and storage requirements for full nodes, potentially pricing out ordinary users and concentrating nodes.
- **Users Trust Miners:** Fewer full nodes mean more reliance on miners for transaction validation, undermining Bitcoin's trustless nature.
- **Validation Difficulty:** Users with limited resources might struggle to validate the entire blockchain or audit the total Bitcoin supply.
- **UTXO Set Growth:** The Unspent Transaction Output (UTXO) set would grow faster, increasing node operational costs.
- **Slower Block Propagation:** Larger blocks take longer to propagate, potentially increasing orphaned blocks and reducing mining decentralization.
- **Consensus Change (Hard Fork):** Non-backward-compatible block size increases require a contentious hard fork, risking chain splits.

The debate led to proposals like Bitcoin XT, Bitcoin Classic, and Bitcoin Unlimited. In 2017, Segregated Witness (SegWit) effectively increased the block capacity (to around 4MB of block weight), implemented as a User Activated Soft Fork (UASF). The failure of the SegWit2x agreement (aiming for a further block size increase) led to the Bitcoin Cash (BCH) hard fork, which adopted larger blocks.

3 Layer 2 and Off-Chain Solutions

Given the challenges of on-chain scaling, significant effort has focused on Layer 2 and off-chain solutions. These aim to increase transaction throughput without directly altering the Bitcoin base layer or by using it more efficiently.

3.1 Payment Channels: The Lightning Network

Proposed by Joseph Poon and Thaddeus Dryja in 2015, the Lightning Network enables off-chain payments.

- **Mechanism:** Uses 2-party payment channels with Hashed TimeLock Contracts (HTLCs). Parties transact multiple times off-chain, settling the net result on-chain when the channel closes. Payments can be routed across a network of channels. "Justice Transactions" penalize cheating attempts.
- **Pros:** Instant finality, high scalability, cheap fees, live on mainnet, trustless dispute resolution.

- **Cons:** User complexity, onboarding/inbound liquidity challenges, need to monitor the chain (or use watchtowers), typically need to be online to receive, liquidity management difficulties, doesn't scale with UTXO ownership on-chain, can push users to custodial wallets.

3.2 Sidechains

Sidechains are separate blockchains pegged to a mainchain (Bitcoin), allowing assets to be transferred between them. They have their own consensus mechanism and a bridge to Bitcoin.

- **Building Blocks:**
  - *Consensus Mechanisms:* Federated, Merge Mining, Proof of Burn, Bitcoin Staking.
  - *Bridge Mechanisms:* Federated, 1-way peg, BitVM Bridge.
  - *Potential Bitcoin Consensus Changes for Sidechains:* BIP300, BigInt arithmetic, OP_CAT.
- **Example: Liquid Network** (Blockstream, 2017)
  - *Mechanism:* Fork of Bitcoin codebase, federated consensus, federated bridge (HSMs).
  - *Pros:* Good UX, cheap fees, basic privacy (Confidential Transactions), allows innovation, interoperable with Lightning, auditable, live.
  - *Cons:* Fully trusted (relies on federation), small user base, federation-managed interoperability.
- **Example: Drivechains** (Paul Sztorc, BIP300, 2015)
  - *Mechanism:* Merge mining for consensus, "Hashrate Escrow" for bridge (miners vote on sidechain state for withdrawals).
  - *Pros:* Potential for good UX, cheap fees, fosters innovation, pays fees to Bitcoin miners, Lightning compatible.
  - *Cons:* Relies on trusting miner incentives, potential for effective block size increase if sidechain data posted on-chain, requires soft fork on Bitcoin.

3.3 Rollups

Rollups execute transactions off-chain but post transaction data (or state diffs) to the mainchain, using the mainchain for consensus and data availability. They prove state transition validity on the mainchain (e.g., ZK-proofs, optimistic rollups). Explored for Bitcoin around 2023.

- **Pros:** Simple for users, easy onboarding, significant (e.g., 20x) on-chain throughput increase, high expressiveness.
- **Cons:** Data availability challenges, technological complexity, governance keys (centralization risk), bridge security, sequencer election. Many Bitcoin designs are in test phase.

3.4 E-Cash Models

These focus on scalable and private digital cash.

- **Chaumian E-cash (Federated Mints)** (Based on David Chaum, 1982)
  - *Mechanism:* A mint (or federation of mints like Fedimint (2021), Cashu (2022)) issues tokens representing BTC, using blind signatures for privacy.
  - *Pros:* Great UX, easy onboarding, scales extremely well, very cheap fees, instant finality, good privacy, Lightning interoperable, simple technology.
  - *Cons:* Fully trusted (relies on mint/federation), not externally auditable, potential regulatory issues, complex cross-mint transfers.
- **zkCoins ("Trustless E-cash")** (Robin Linus, 2023)
  - *Mechanism:* Client-side validation using Zero-Knowledge Proofs (ZKPs). Users verify off-chain; mainchain used for double-spend prevention via small commitments.
  - *Pros:* Potential for throughput increase (e.g., 10x), strong privacy, trustless, cheap transactions, easy onboarding, Lightning interoperable, pays fees to Bitcoin miners.
  - *Cons:* High complexity, bridge mechanism needed if a separate layer, crucial user data backups, many implementations in test phase.

3.5 Shared UTXOs

Multiple users share control or interest in a single UTXO on the Bitcoin blockchain.

- **Statechains** (Ruben Somsen, 2019)
  - *Mechanism:* Shared UTXO with client-server model. Server co-signs transactions with a one-time key for off-chain ownership transfer. Unilateral exit allows owner to claim funds on-chain.
  - *Pros:* Good UX, cheap fees, great scalability, Lightning compatible, forward integrity, some implementations live.
  - *Cons:* Trust server not to collude/censor (mitigated by unilateral exit), users need to watch chain, potential regulatory issues, off-chain transfers don't pay miners.
- **Ark** (Burak, 2023)
  - *Mechanism:* Shared UTXO model, client-server architecture. Servers issue Virtual UTXOs (VTXOs) with limited lifetime for off-chain transactions. Unilateral exit.
  - *Pros:* Simple for users, easy onboarding, scales well, Lightning interoperable.
  - *Cons:* Users need to be online (or use service), significant liquidity for Ark providers, potential bootstrapping problems, complex cross-Ark payments, test phase, VTXO transfers don't pay miners, might need Bitcoin consensus change for optimal implementation.

4 Light Clients in Bitcoin

Satoshi Nakamoto's initial vision for Bitcoin acknowledged that not all users would need to run full nodes. Instead, users could utilize **light clients** (also known as Simplified Payment Verification or SPV clients) or pruned full nodes.

- **Functionality:** Light clients download and verify block headers from full nodes. They confirm that proof-of-work has been done and that the chain of headers is the longest (most-work) chain among the ones they received.

- **Transaction Verification:** To verify if a transaction is included in a block, a light client requests a Merkle proof from a full node. This proof consists of the transaction itself and the (siblings of the) "branch" of the Merkle tree connecting the transaction to the Merkle root in the block header. The client can then verify this proof against the Merkle root in the validated block header.

- **Benefits:** This allows users to verify their own transactions with significantly less resource usage (storage, bandwidth) compared to running a full node, thereby enhancing usability.

- **Trust Assumptions:** Light clients trust that the longest chain of headers they receive is the honest chain (i.e., at least one of its connections is an honest node). Security can be increased by connecting to multiple full nodes.

Pruned full nodes verify all blocks and transactions but then discard older block data (except headers) to save space, still contributing to network security but with limited ability to serve historical data to new nodes.

5 The Ongoing Quest for Scalability and Trade-offs

The journey to scale Bitcoin is ongoing, with no single "free lunch" solution. Every proposed method involves trade-offs between scalability, decentralization, trust assumptions, complexity, and user experience.

A conceptual visualization (often depicted in diagrams within the community, for example, as described by reference to an image IMG_2933.JPG in the source material) might plot various solutions on axes of "Scalability" and "Decentralization."

- "Big Blocks" might offer higher raw transaction throughput but at the cost of decentralization due to increased node requirements.

- "Lightning Network" aims for high scalability with better decentralization compared to naive block size increases. However, it has some operational difficulties (e.g., the liquidity problem).

- Emerging solutions like "Rollups" and "zkCoins" project significant throughput multipliers (e.g., 10x-20x).

- Such solutions often include specific mechanisms, like shared UTXO models (e.g., "UTXO 10 BTC 4-of-4 Server" branching to user VTXOs), highlighting the complex interactions and resource management involved.

The Bitcoin community continues to research and develop solutions across paradigms like payment channels, sidechains, shared UTXOs, and e-cash models. The core challenge persists: enabling Bitcoin to serve a global user base efficiently and affordably while preserving its foundational properties as a peer-to-peer electronic cash system, a concern that has driven innovation since its earliest days.