1. Introduction

- Existing online payments rely on financial institutions, which act as intermediaries.
- This trust-based model leads to costs, transaction reversals, and fraud.
- A **peer-to-peer (P2P) electronic cash system** can solve these issues.
- The proposed system eliminates the need for a trusted third party by using cryptographic proof.

2. Transactions

- Bitcoin transactions use a **chain of digital signatures**.
- Each owner transfers coins by signing the previous transaction and the recipient's public key.
- The challenge is preventing **double-spending** without a central authority.

3. Timestamp Server

- Transactions are grouped into blocks and **timestamped**.
- Each block references the previous one, creating an immutable chain.
- This forms a chronological record of transactions.

4. Proof-of-Work (PoW)

- Uses **computational puzzles** (like Hashcash) to secure the blockchain.
- Miners must find a hash with a required number of leading zeros.
- The longest chain represents the most CPU effort and is accepted as the valid chain.
- **Modifying past transactions requires redoing all the work, making attacks impractical**.

5. Network Protocol

1. Transactions are broadcasted to all nodes.
2. Nodes collect transactions into blocks.
3. Miners compete to find a valid proof-of-work.
4. The new block is broadcasted and verified.
5. Nodes accept the longest valid chain.

6. Incentives

- Miners receive rewards in the form of **newly created bitcoins and transaction fees**.
- This incentivizes network security and prevents malicious activity.

7. Disk Space Optimization

- Transactions are stored in a **Merkle Tree**, allowing old transactions to be pruned.
- Only block headers (80 bytes each) need to be kept, reducing storage requirements.

8. Simplified Payment Verification (SPV)

- Users can verify transactions without running a full node.
- SPV relies on the **longest blockchain** and Merkle proofs.

9. Combining and Splitting Transactions

- Bitcoin allows transactions with **multiple inputs and outputs**.
- This enables efficient transfers and handling of change.

10. Privacy

- Unlike banks, Bitcoin does not store identities, only public keys.
- Privacy is maintained as **transactions are pseudonymous**.
- However, multi-input transactions can reveal ownership links.

11. Security and Attack Analysis

- An attacker trying to rewrite history must outpace the honest chain.
- The probability of success decreases **exponentially** as more blocks are added.
- This makes Bitcoin secure as long as the majority of computing power is honest.

12. Conclusion

- Bitcoin provides a decentralized, trustless system for digital payments.
- Proof-of-work ensures security and prevents fraud.
- Consensus is achieved by **CPU-based voting**, making the system resilient.