Bitcoin

Saravanan Vijayakumaran
sarva@ee.iitb.ac.in

Department of Electrical Engineering
Indian Institute of Technology Bombay

July 30, 2019

What is Bitcoin?

What is Bitcoin?

- Cryptocurrency

What is Bitcoin?

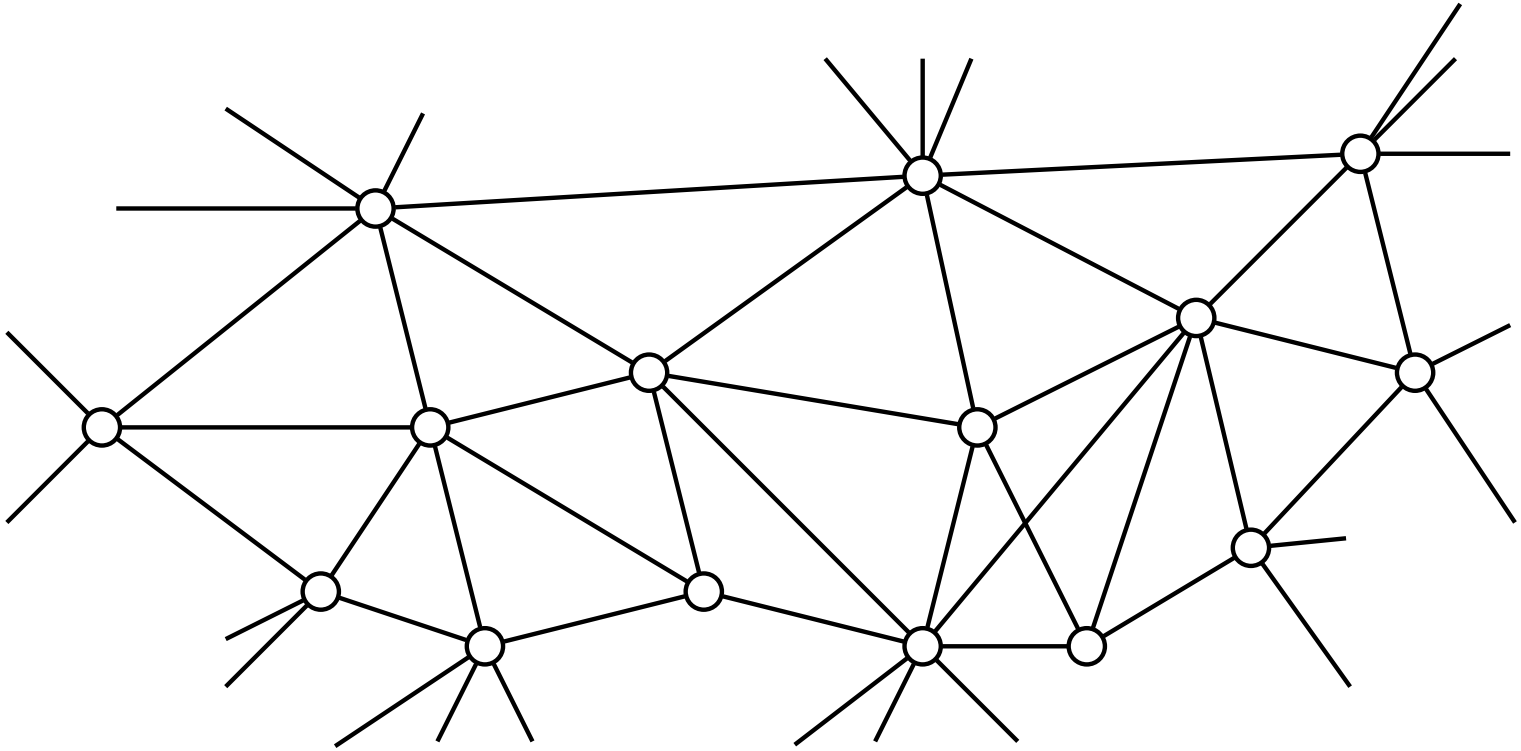
- Cryptocurrency
- Open source

What is Bitcoin?

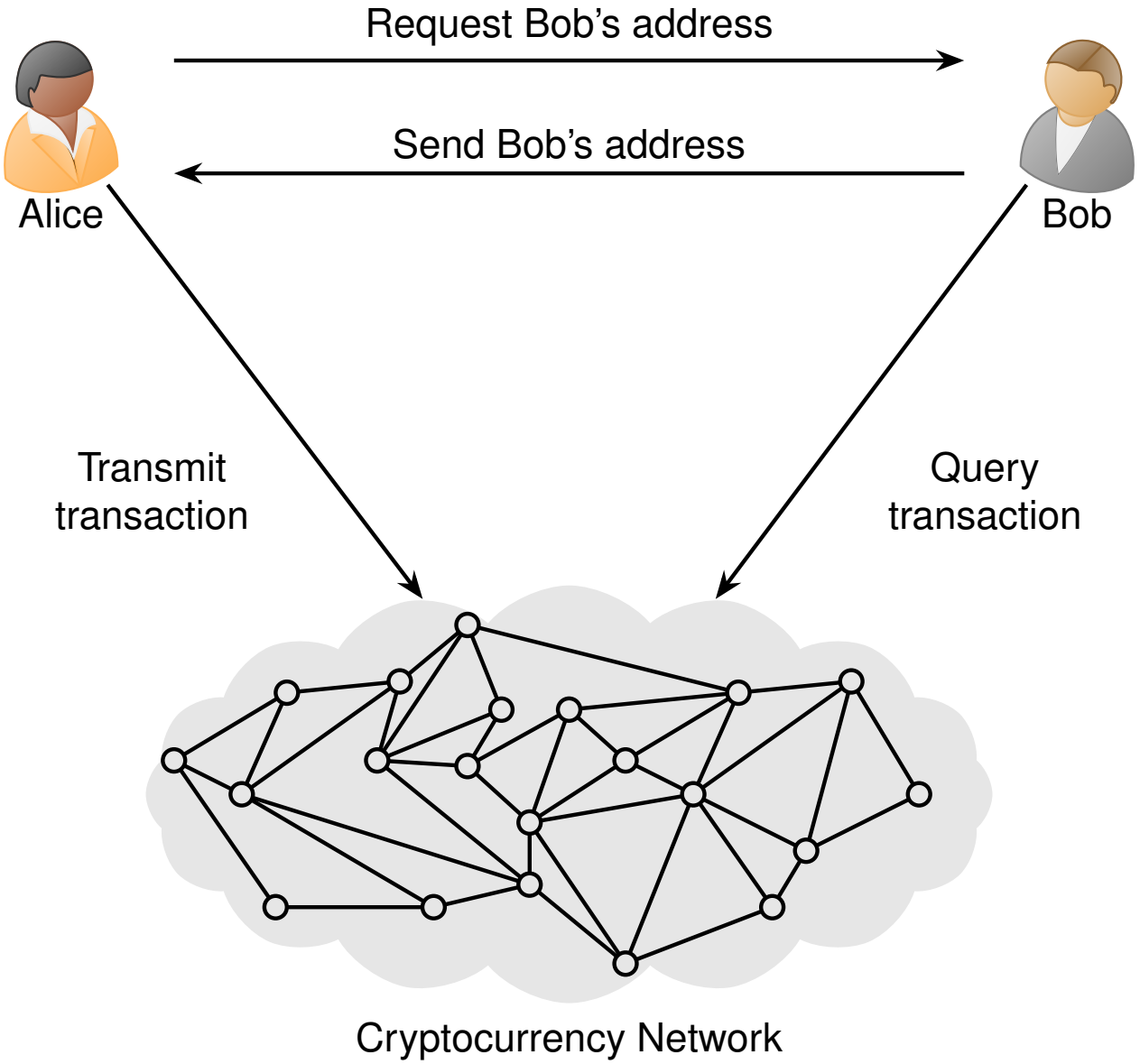
- Cryptocurrency
- Open source
- Decentralized network

What is Bitcoin?

- Cryptocurrency
- Open source
- Decentralized network



Cryptocurrency Transaction Workflow



Decentralization Challenges

Decentralization Challenges

- Counterfeiting

Decentralization Challenges

- Counterfeiting
- Currency creation rules

Decentralization Challenges

- Counterfeiting
- Currency creation rules
- Double spending

Decentralization Challenges

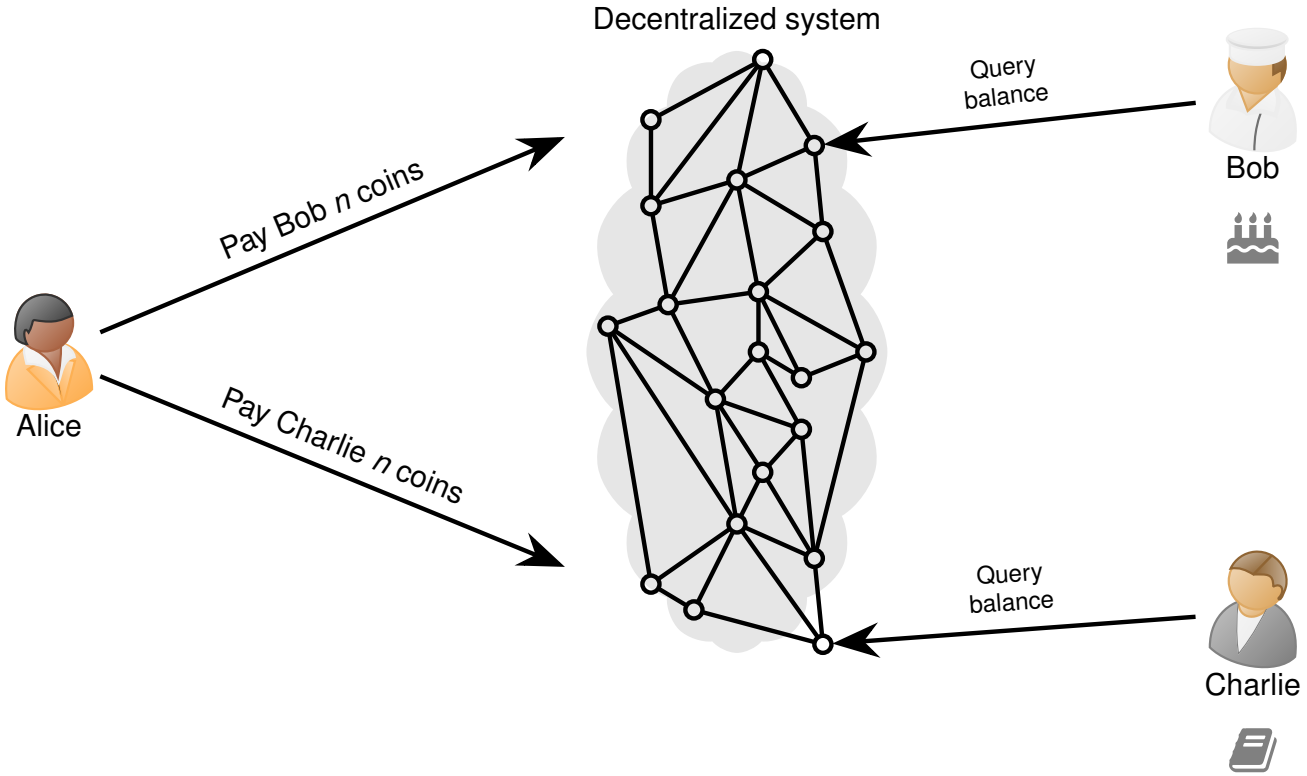
- Counterfeiting
- Currency creation rules
- Double spending
  - Alice pays Bob n digicoins for a cake

Decentralization Challenges

- Counterfeiting
- Currency creation rules
- Double spending
  - Alice pays Bob *n* digicoins for a cake
  - Alice uses the **same** *n* digicoins to pay Charlie for a book

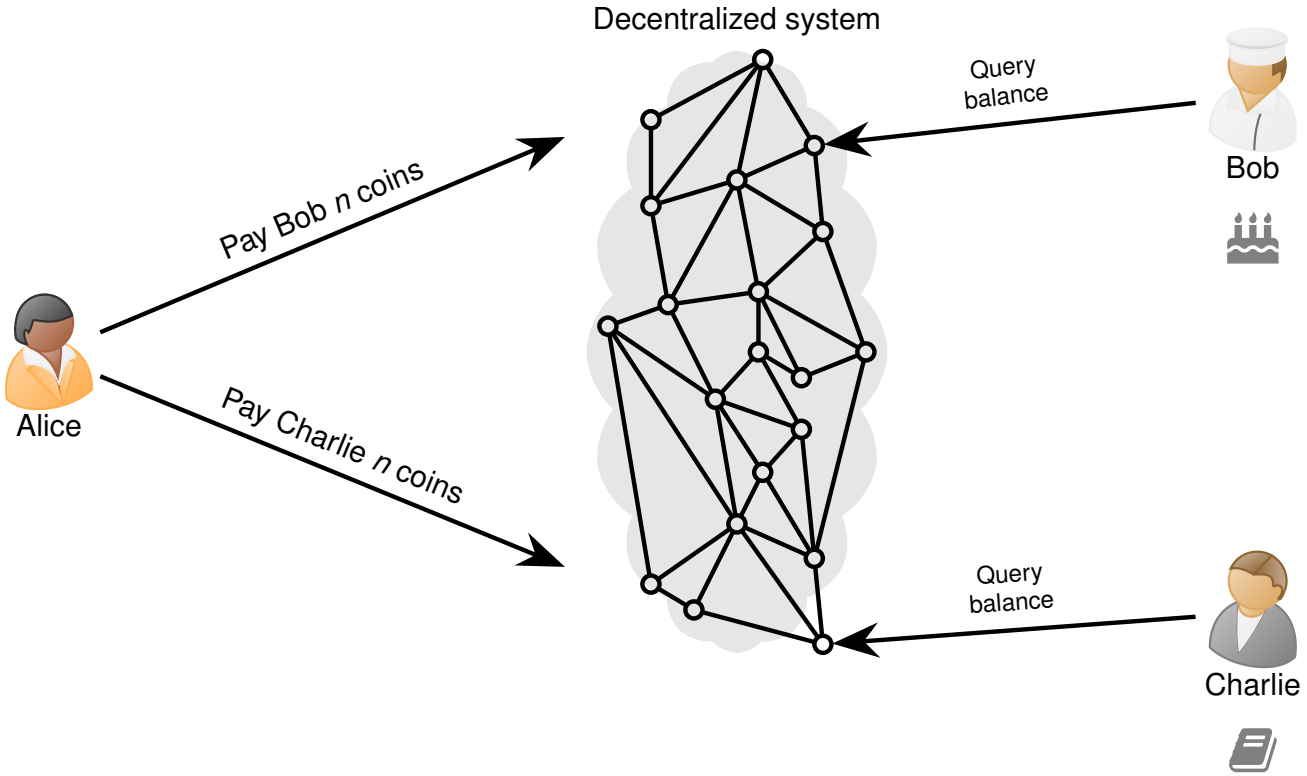
Decentralization Challenges

- Counterfeiting
- Currency creation rules
- Double spending
  - Alice pays Bob *n* digicoins for a cake
  - Alice uses the **same** *n* digicoins to pay Charlie for a book



Decentralization Challenges

- Counterfeiting
- Currency creation rules
- Double spending
  - Alice pays Bob *n* digicoins for a cake
  - Alice uses the **same** *n* digicoins to pay Charlie for a book



Solution without a central coordinator?

Double Spending

Double Spending

- Familiar to academics

Double Spending

- Familiar to academics
- Submitting same paper to two conferences

Double Spending

- Familiar to academics
- Submitting same paper to two conferences
- **Possible solution**

Double Spending

- Familiar to academics
- Submitting same paper to two conferences
- **Possible solution**
  Reviewers google paper contents to find duplicates

Double Spending

- Familiar to academics
- Submitting same paper to two conferences
- **Possible solution**
  Reviewers google paper contents to find duplicates
- Solution fails if

Double Spending

- Familiar to academics
- Submitting same paper to two conferences
- **Possible solution**
  Reviewers google paper contents to find duplicates
- Solution fails if
  - Conferences accepting papers at same time

Double Spending

- Familiar to academics
- Submitting same paper to two conferences
- **Possible solution**
  Reviewers google paper contents to find duplicates
- Solution fails if
  - Conferences accepting papers at same time
  - Conference proceedings not published/indexed

Double Spending

- Familiar to academics
- Submitting same paper to two conferences
- **Possible solution**
  Reviewers google paper contents to find duplicates
- Solution fails if
  - Conferences accepting papers at same time
  - Conference proceedings not published/indexed
- **Better solution**

Double Spending

- Familiar to academics
- Submitting same paper to two conferences
- **Possible solution**
  Reviewers google paper contents to find duplicates
- Solution fails if
  - Conferences accepting papers at same time
  - Conference proceedings not published/indexed
- **Better solution**
  A single public database to store all submissions to all conferences

The Blockchain

The Blockchain

Blockchain:

The Blockchain

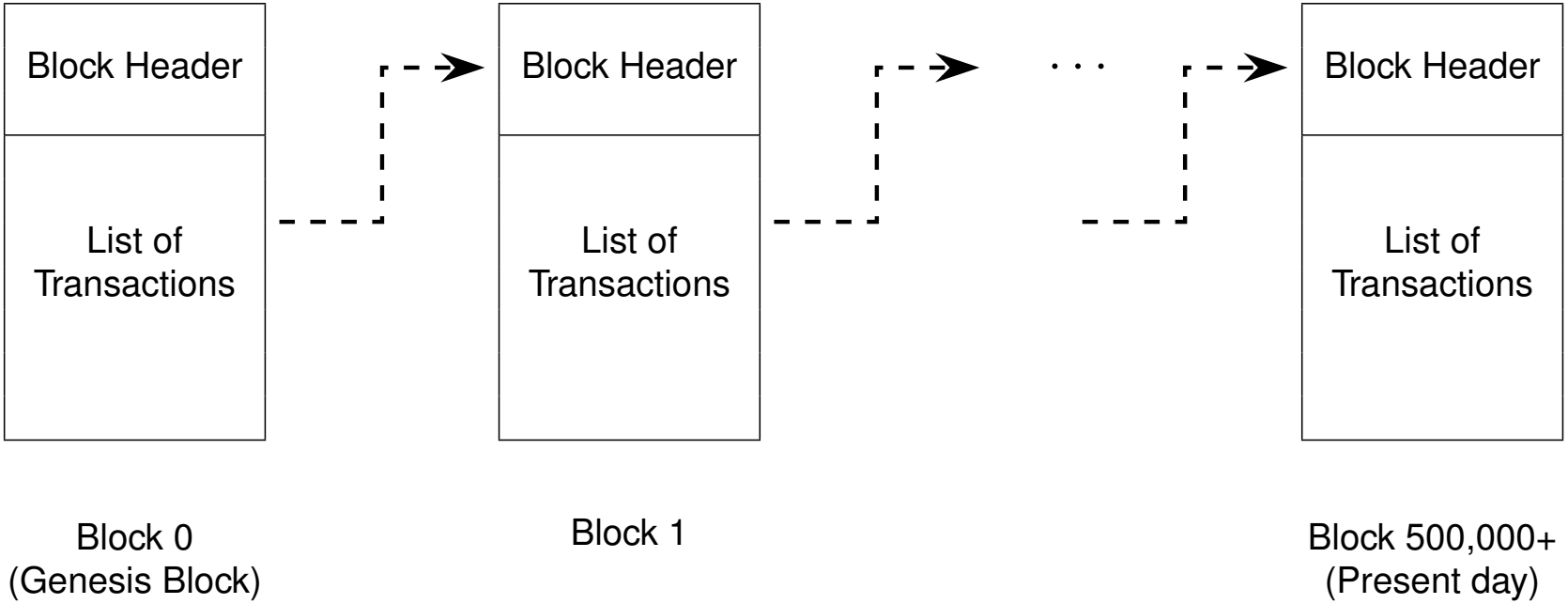
Blockchain: A public database to store all transactions

The Blockchain

Blockchain: A public database to store all transactions which is replicated by many network nodes

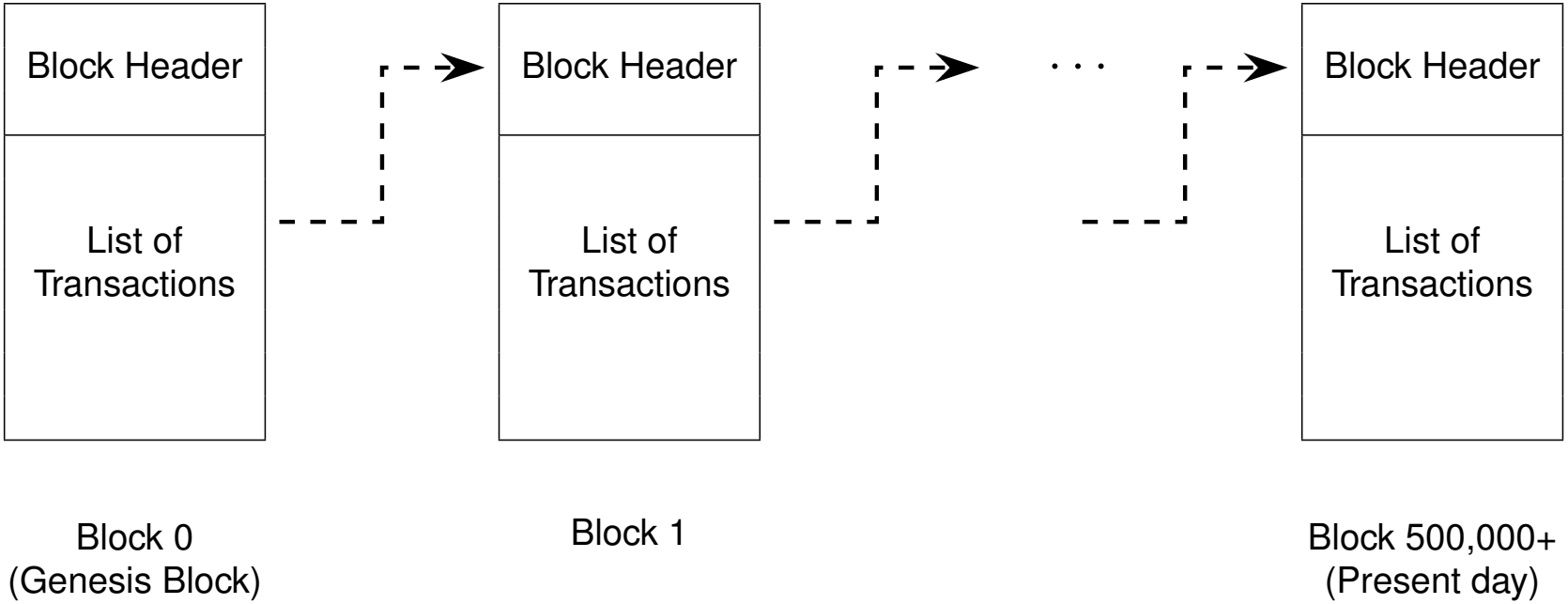
The Blockchain

Blockchain: A public database to store all transactions which is replicated by many network nodes



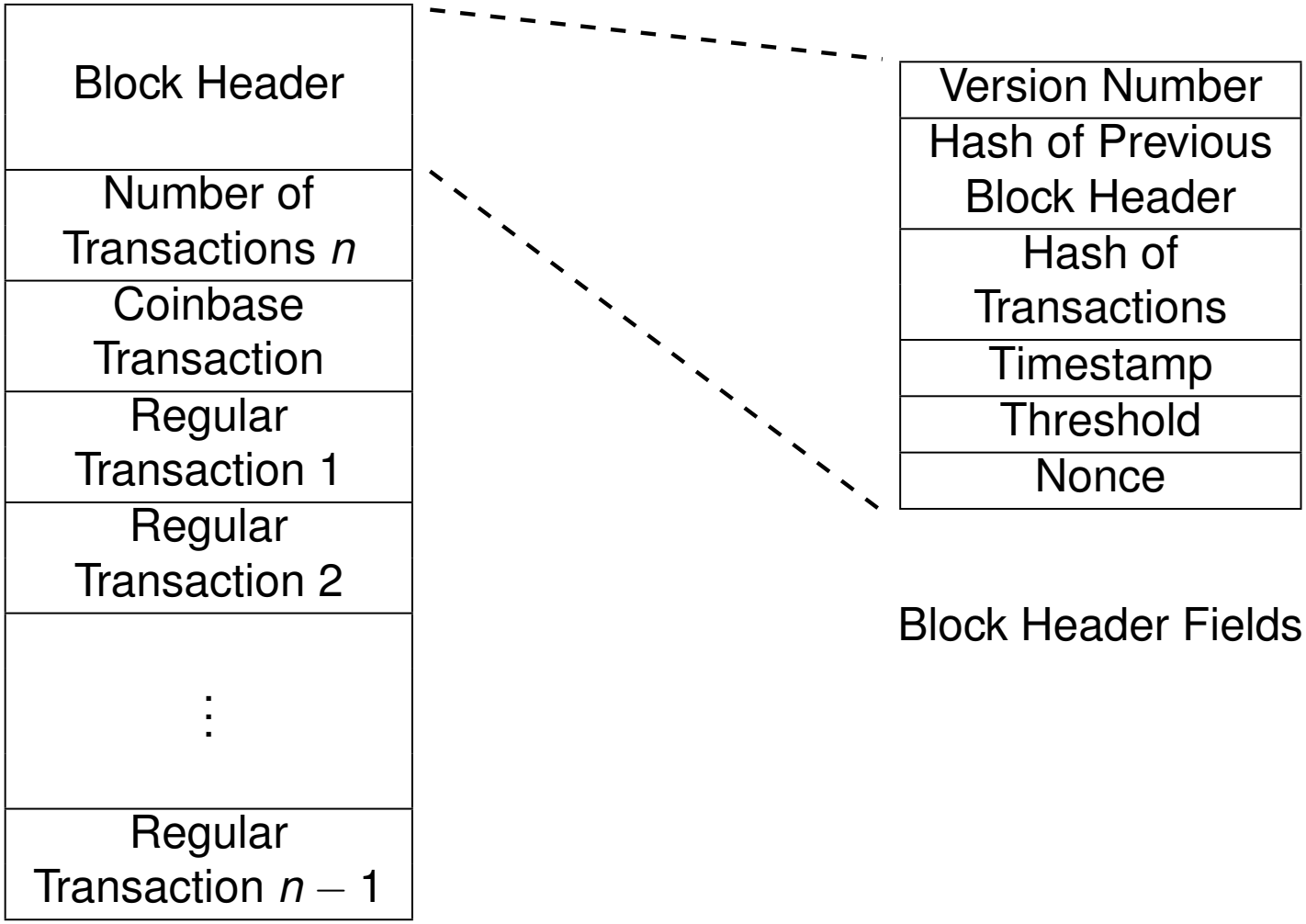
The Blockchain

Blockchain: A public database to store all transactions which is replicated by many network nodes



How are the blocks linked?

Bitcoin Block and Header Formats



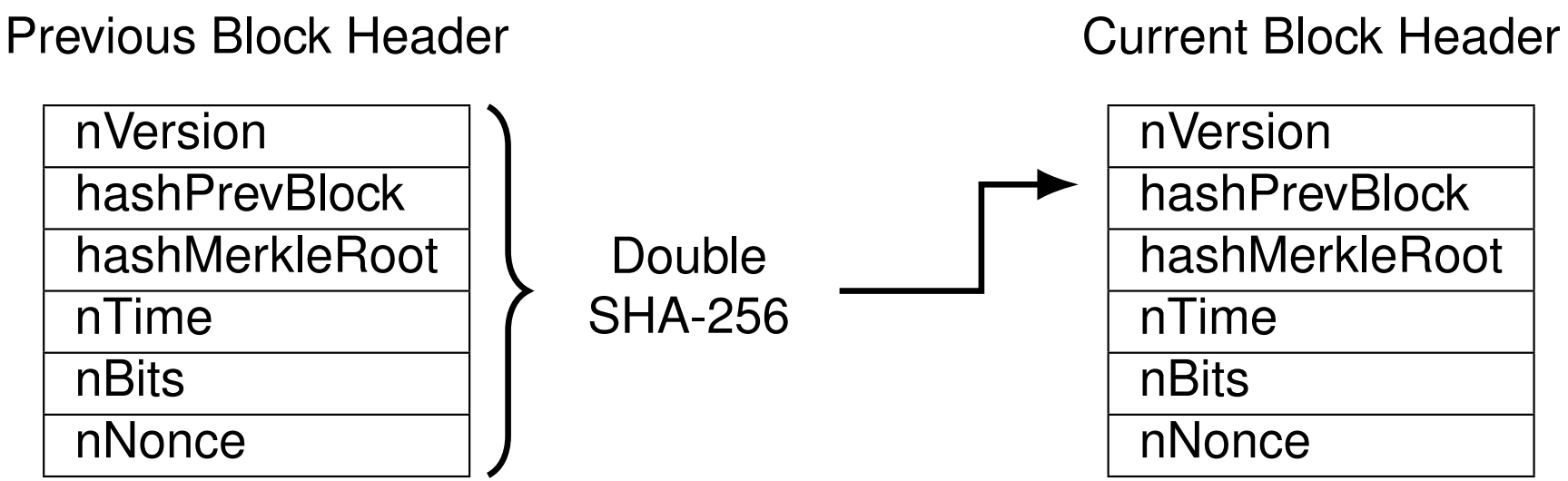
- Hash = Output of cryptographic hash function

Block Header

nVersion	4 bytes
hashPrevBlock	32 bytes
hashMerkleRoot	32 bytes
nTime	4 bytes
nBits	4 bytes
nNonce	4 bytes

Block Header

nVersion	4 bytes
hashPrevBlock	32 bytes
hashMerkleRoot	32 bytes
nTime	4 bytes
nBits	4 bytes
nNonce	4 bytes



Cryptographic Hash Functions

- Easy to compute but difficult to invert
- Collision-resistant
- Pseudorandom outputs

Cryptographic Hash Functions

- Easy to compute but difficult to invert
- Collision-resistant
- Pseudorandom outputs
- SHA-256 = NIST approved CHF with 256-bit outputs

Cryptographic Hash Functions

- Easy to compute but difficult to invert
- Collision-resistant
- Pseudorandom outputs
- SHA-256 = NIST approved CHF with 256-bit outputs

Input	SHA-256 Output
july0	171c9f5053d5d675d1d1ed477c908e98498e6751ae392a78807c3cd6ad6975fa
july1	7d8033d140d8b8db8324753a25c5e32ee4faa9c4e306bddb317907be51cd8a24
july2	bda0b2ab2c7d654589b32f46a548cba27b7371f27b070ddd7d3b87122a078f06
july3	dfa3569a46b1a13c24c9f385da140f4763a3fbb70f8eebe0f29ba535145d32ca
july4	27d39d26edc54c11cc78d17bf0dd294413300dd004127fa6dcff368ea74bb87c
july5	a0ebd3e23823fc291b090abd2eb1403912be6b72398f3bf4e92c4ec555902d53
july6	dc7d6bcc266af402e53b9fb978b6579940bb97743f6e975a988cb20d903e0c5f
july7	984906fbbaa7dbad2ee01a81df7a237bfdb63aeb06b4cf97a89fc004542c1dab
july8	7be4d491b73a4797304980070d5b5fb5c7fd6921e70efc7ce38023c50664803d
july9	e8c4af8895bcddb9cea3e3e1e8a08e090690bb55fd6617da5aa0873f27e218ee

Cryptographic Hash Functions

- Easy to compute but difficult to invert
- Collision-resistant
- Pseudorandom outputs
- SHA-256 = NIST approved CHF with 256-bit outputs

Input	SHA-256 Output
`july0`	`171c9f5053d5d675d1d1ed477c908e98498e6751ae392a78807c3cd6ad6975fa`
`july1`	`7d8033d140d8b8db8324753a25c5e32ee4faa9c4e306bddb317907be51cd8a24`
`july2`	`bda0b2ab2c7d654589b32f46a548cba27b7371f27b070ddd7d3b87122a078f06`
`july3`	`dfa3569a46b1a13c24c9f385da140f4763a3fbb70f8eebe0f29ba535145d32ca`
`july4`	`27d39d26edc54c11cc78d17bf0dd294413300dd004127fa6dcff368ea74bb87c`
`july5`	`a0ebd3e23823fc291b090abd2eb1403912be6b72398f3bf4e92c4ec555902d53`
`july6`	`dc7d6bcc266af402e53b9fb978b6579940bb97743f6e975a988cb20d903e0c5f`
`july7`	`984906fbbaa7dbad2ee01a81df7a237bfdb63aeb06b4cf97a89fc004542c1dab`
`july8`	`7be4d491b73a4797304980070d5b5fb5c7fd6921e70efc7ce38023c50664803d`
`july9`	`e8c4af8895bcddb9cea3e3e1e8a08e090690bb55fd6617da5aa0873f27e218ee`

- Hex digits: `0 = 0000, 1 = 0001, 2 = 0010,..., a = 1010, b = 1011, c = 1100,..., e = 1110, f = 1111`

Cryptographic Hash Functions

- Easy to compute but difficult to invert
- Collision-resistant
- Pseudorandom outputs
- SHA-256 = NIST approved CHF with 256-bit outputs

Input	SHA-256 Output
`july0`	`171c9f5053d5d675d1d1ed477c908e98498e6751ae392a78807c3cd6ad6975fa`
`july1`	`7d8033d140d8b8db8324753a25c5e32ee4faa9c4e306bddb317907be51cd8a24`
`july2`	`bda0b2ab2c7d654589b32f46a548cba27b7371f27b070ddd7d3b87122a078f06`
`july3`	`dfa3569a46b1a13c24c9f385da140f4763a3fbb70f8eebe0f29ba535145d32ca`
`july4`	`27d39d26edc54c11cc78d17bf0dd294413300dd004127fa6dcff368ea74bb87c`
`july5`	`a0ebd3e23823fc291b090abd2eb1403912be6b72398f3bf4e92c4ec555902d53`
`july6`	`dc7d6bcc266af402e53b9fb978b6579940bb97743f6e975a988cb20d903e0c5f`
`july7`	`984906fbbaa7dbad2ee01a81df7a237bfdb63aeb06b4cf97a89fc004542c1dab`
`july8`	`7be4d491b73a4797304980070d5b5fb5c7fd6921e70efc7ce38023c50664803d`
`july9`	`e8c4af8895bcddb9cea3e3e1e8a08e090690bb55fd6617da5aa0873f27e218ee`

- Hex digits: `0 = 0000, 1 = 0001, 2 = 0010,..., a = 1010, b = 1011, c = 1100,..., e = 1110, f = 1111`
- At a billion outputs per second, 78 billion years required to calculate 2^{100} outputs

Hashcash

Hashcash

- A database you own where anyone in the world can add entries?

Hashcash

- A database you own where anyone in the world can add entries? Your email inbox

Hashcash

- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam

Hashcash

- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam
- Protocol

Hashcash

- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam
- Protocol
  - Suppose an email client wants to send email to an email server

Hashcash

- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam
- Protocol
  - Suppose an email client wants to send email to an email server
  - Client and server agree upon a cryptographic hash function H

Hashcash

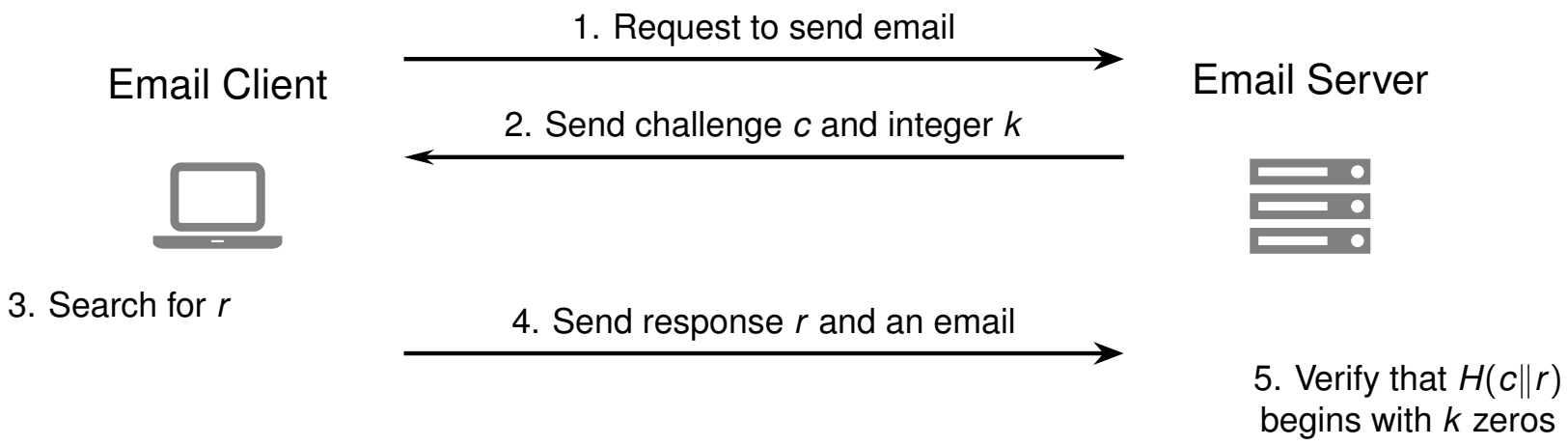
- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam
- Protocol
  - Suppose an email client wants to send email to an email server
  - Client and server agree upon a cryptographic hash function H
  - Email server sends the client a challenge string c

Hashcash

- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam
- Protocol
  - Suppose an email client wants to send email to an email server
  - Client and server agree upon a cryptographic hash function H
  - Email server sends the client a challenge string c
  - Client needs to find a string r such that $H(c \| r)$ begins with k zeros

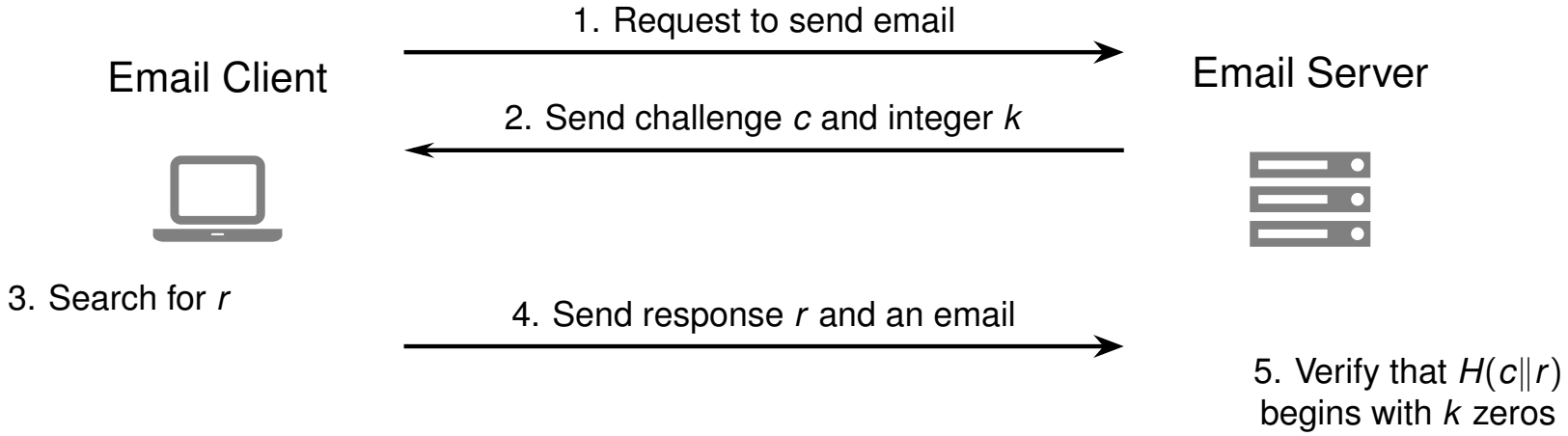
Hashcash

- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam
- Protocol
  - Suppose an email client wants to send email to an email server
  - Client and server agree upon a cryptographic hash function H
  - Email server sends the client a challenge string c
  - Client needs to find a string r such that $H(c \| r)$ begins with k zeros



Hashcash

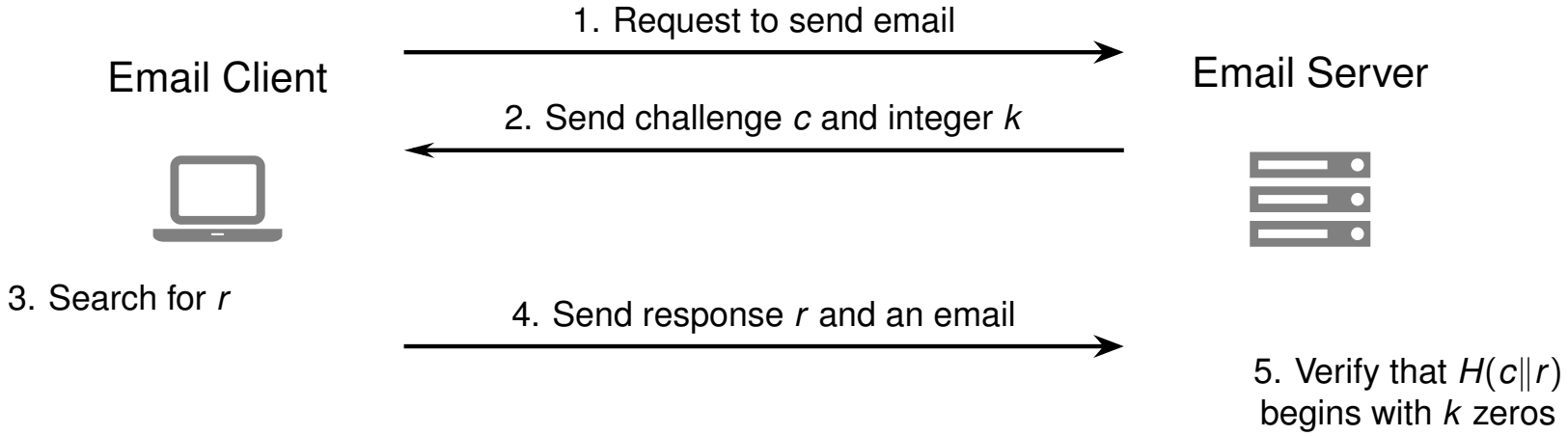
- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam
- Protocol
  - Suppose an email client wants to send email to an email server
  - Client and server agree upon a cryptographic hash function H
  - Email server sends the client a challenge string c
  - Client needs to find a string r such that $H(c||r)$ begins with k zeros



- The r is considered **proof-of-work (PoW)**; difficult to generate but easy to verify

Hashcash

- A database you own where anyone in the world can add entries? Your email inbox
- Hashcash was proposed in 1997 to prevent spam
- Protocol
  - Suppose an email client wants to send email to an email server
  - Client and server agree upon a cryptographic hash function H
  - Email server sends the client a challenge string c
  - Client needs to find a string r such that $H(c||r)$ begins with k zeros



- The r is considered **proof-of-work (PoW)**; difficult to generate but easy to verify
- Demo

Difficulty Increases with k

- Let hash function output length n be 4 bits

Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Binary	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111

$k = 3$

$k = 2$

$k = 1$

Difficulty Increases with k

- Let hash function output length n be 4 bits

Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Binary	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111

$k = 3$

$k = 2$

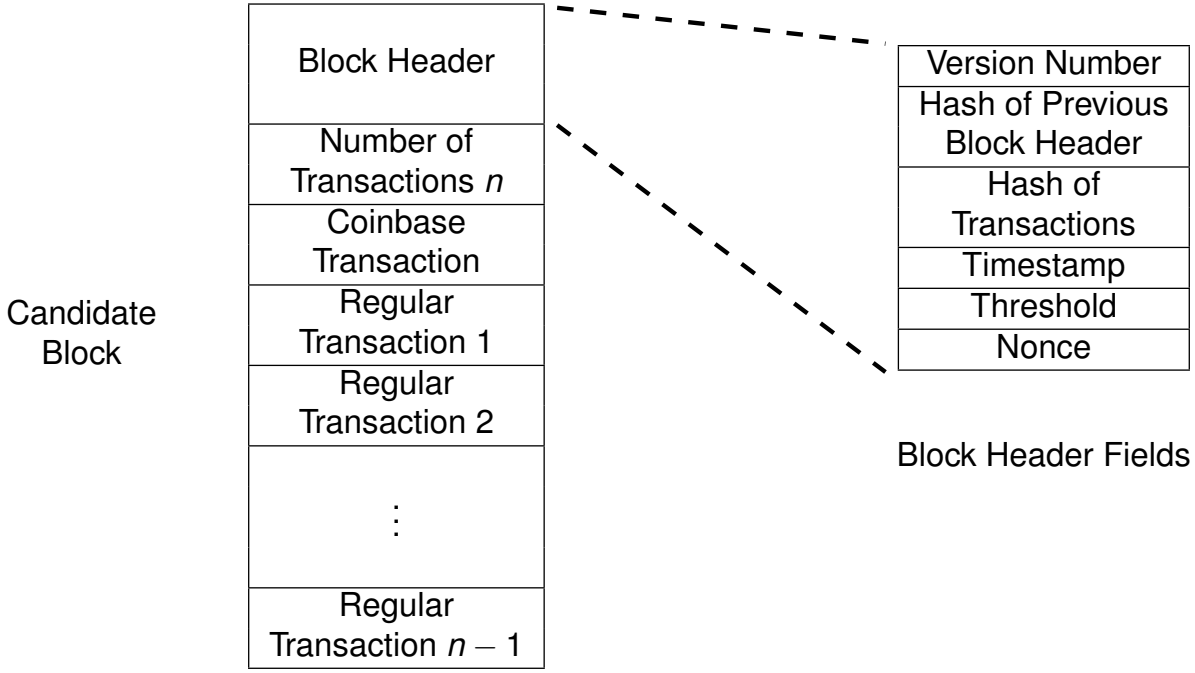
$k = 1$

- Since H has pseudorandom outputs, probability of success in a single trial is

$$\frac{2^{n-k}}{2^n} = \frac{1}{2^k}$$

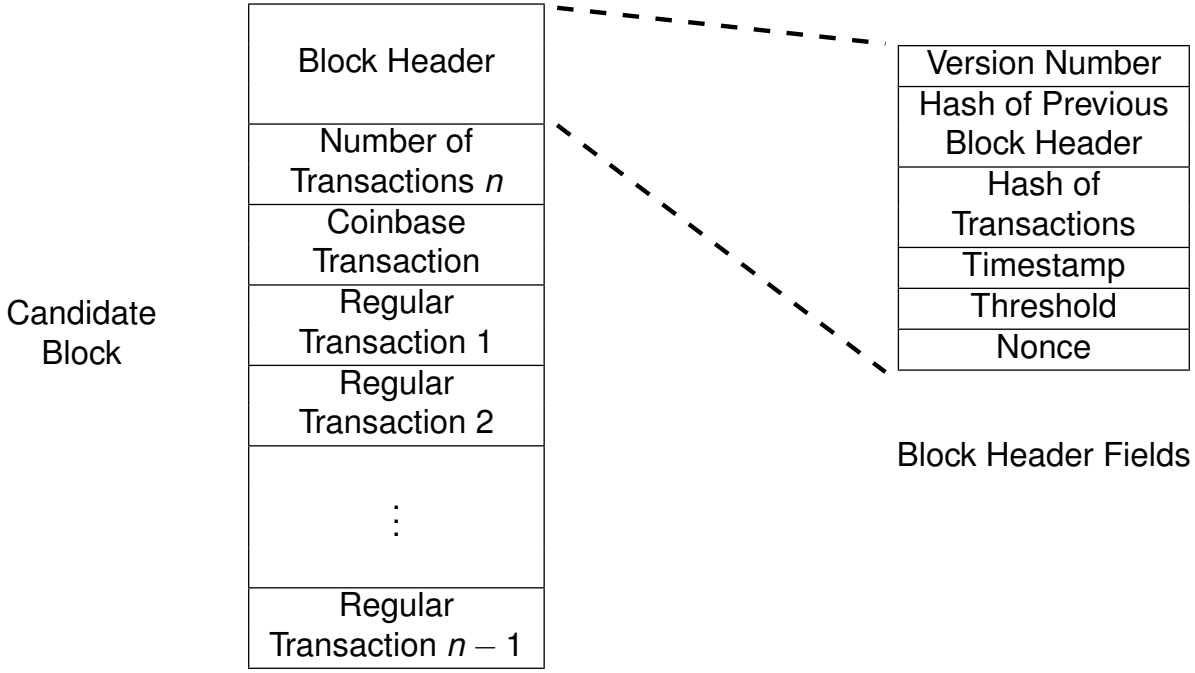
Bitcoin Mining

- Mining = Process of adding new blocks to the blockchain
- Nodes which want to perform transactions broadcast them
- Miners collect some of these transactions into a candidate block



Bitcoin Mining

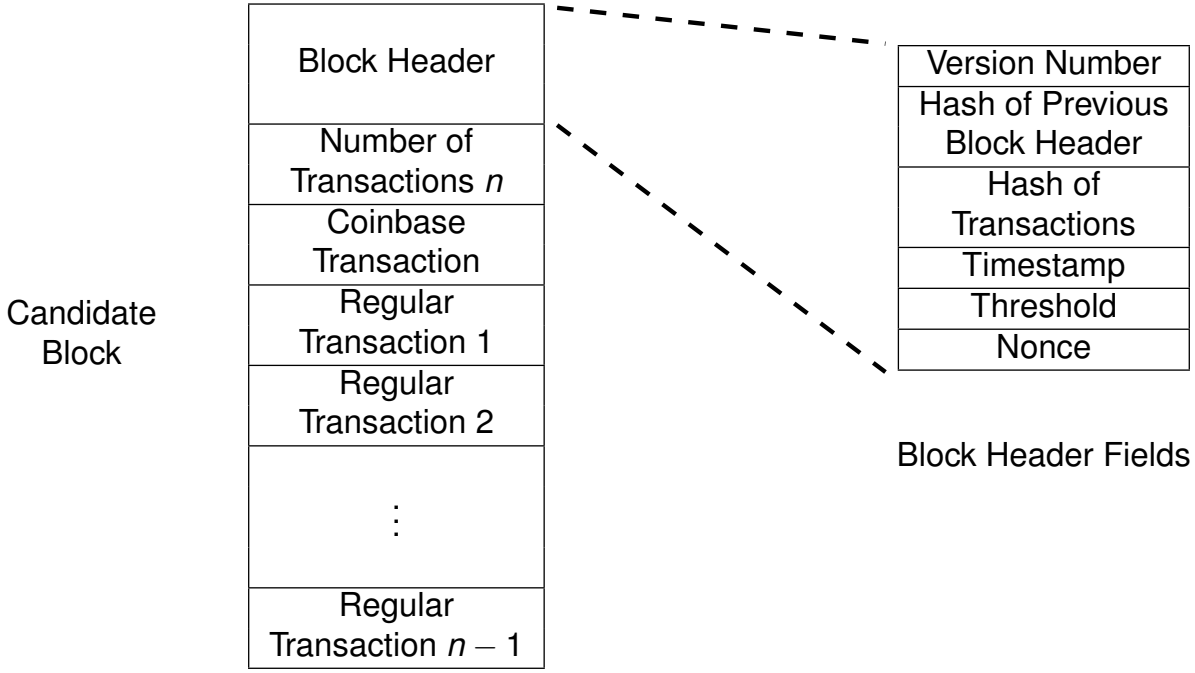
- Mining = Process of adding new blocks to the blockchain
- Nodes which want to perform transactions broadcast them
- Miners collect some of these transactions into a candidate block



- Threshold encodes a 256-bit value like $0x\underbrace{00\cdots00}_{16\text{ times}}\underbrace{FFFFF\cdots FFFFF}_{48\text{ times}}$

Bitcoin Mining

- Mining = Process of adding new blocks to the blockchain
- Nodes which want to perform transactions broadcast them
- Miners collect some of these transactions into a candidate block



- Threshold encodes a 256-bit value like $0x\underbrace{00\cdots00}_{\text{16 times}}\underbrace{FFFFF\cdots FFFFF}_{\text{48 times}}$
- Miner who can find Nonce such that

$$\text{SHA256}(\text{SHA256}(\underbrace{\text{Version Number} \| \cdots \| \text{Nonce}}_{\text{Candidate Block Header}})) \leq \text{Threshold}.$$

can add a new block

Mining is Hard

Mining is Hard

Target value T	Fraction of SHA256d outputs $\leq T$
$0x7\underbrace{FFFF\ FFFF \cdots FFFF}_{63 \text{ times}}$	

Mining is Hard

Target value T	Fraction of SHA256d outputs $\leq T$
$0x7\underbrace{FFFF\ FFFF \cdots FFFF}_{63 \text{ times}}$	$\frac{1}{2}$

Mining is Hard

Target value T	Fraction of SHA256d outputs $\leq T$
$\text{0x7}\underbrace{\text{FFFF FFFF} \cdots \text{FFFF}}_{\text{63 times}}$	$\frac{1}{2}$
$\text{0x0}\underbrace{\text{FFFF FFFF} \cdots \text{FFFF}}_{\text{63 times}}$	

Mining is Hard

Target value T	Fraction of SHA256d outputs $\leq T$
$0x7\underbrace{FFFF\ FFFF \cdots FFFF}_{63 \text{ times}}$	$\frac{1}{2}$
$0x0\underbrace{FFFF\ FFFF \cdots FFFF}_{63 \text{ times}}$	$\frac{1}{16}$

Mining is Hard

Target value T	Fraction of SHA256d outputs $\leq T$
$\text{0x7}\underbrace{\text{FFFF FFFF} \cdots \text{FFFF}}_{\text{63 times}}$	$\frac{1}{2}$
$\text{0x0}\underbrace{\text{FFFF FFFF} \cdots \text{FFFF}}_{\text{63 times}}$	$\frac{1}{16}$
$\text{0x}\underbrace{\text{00} \cdots \text{00}}_{\text{16 times}}\underbrace{\text{FFFFF} \cdots \text{FFFFF}}_{\text{48 times}}$	

Mining is Hard

Target value T	Fraction of SHA256d outputs $\leq T$
$\text{0x7}\underbrace{\text{FFFF FFFF} \cdots \text{FFFF}}_{\text{63 times}}$	$\frac{1}{2}$
$\text{0x0}\underbrace{\text{FFFF FFFF} \cdots \text{FFFF}}_{\text{63 times}}$	$\frac{1}{16}$
$\text{0x}\underbrace{\text{00} \cdots \text{00}}_{\text{16 times}}\underbrace{\text{FFFFF} \cdots \text{FFFFF}}_{\text{48 times}}$	$\frac{1}{2^{64}}$

Mining is Hard

Target value T	Fraction of SHA256d outputs $\leq T$
$0x7\underbrace{FFFF\ FFFF \cdots FFFF}_{63\text{ times}}$	$\frac{1}{2}$
$0x0\underbrace{FFFF\ FFFF \cdots FFFF}_{63\text{ times}}$	$\frac{1}{16}$
$0x\underbrace{00 \cdots 00}_{16\text{ times}}\underbrace{FFFFF \cdots FFFFF}_{48\text{ times}}$	$\frac{1}{2^{64}}$

$$\Pr\left[\text{SHA256d output} \leq T\right] \approx \frac{T+1}{2^{256}}$$

Why should anyone mine blocks?

Why should anyone mine blocks?

- Successful miner gets rewarded in bitcoins

Why should anyone mine blocks?

- Successful miner gets rewarded in bitcoins
- Every block contains a **coinbase transaction** which creates 12.5 bitcoins

Why should anyone mine blocks?

- Successful miner gets rewarded in bitcoins
- Every block contains a **coinbase transaction** which creates 12.5 bitcoins
- Each miner specifies his own address as the destination of the new coins

Why should anyone mine blocks?

- Successful miner gets rewarded in bitcoins
- Every block contains a **coinbase transaction** which creates 12.5 bitcoins
- Each miner specifies his own address as the destination of the new coins
- Every miner is competing to solve their own PoW puzzle

Why should anyone mine blocks?

- Successful miner gets rewarded in bitcoins
- Every block contains a **coinbase transaction** which creates 12.5 bitcoins
- Each miner specifies his own address as the destination of the new coins
- Every miner is competing to solve their own PoW puzzle
- Miners also collect the transaction fees in the block

Mining Farms

- Mining farms have thousands of mining rigs
- Each mining rig has dozens of mining chips
- Each chip has dozens of SHA256 mining cores
- Farms are located in places with cheap power and cooling

Block Addition Workflow

Block Addition Workflow

- Nodes broadcast transactions

Block Addition Workflow

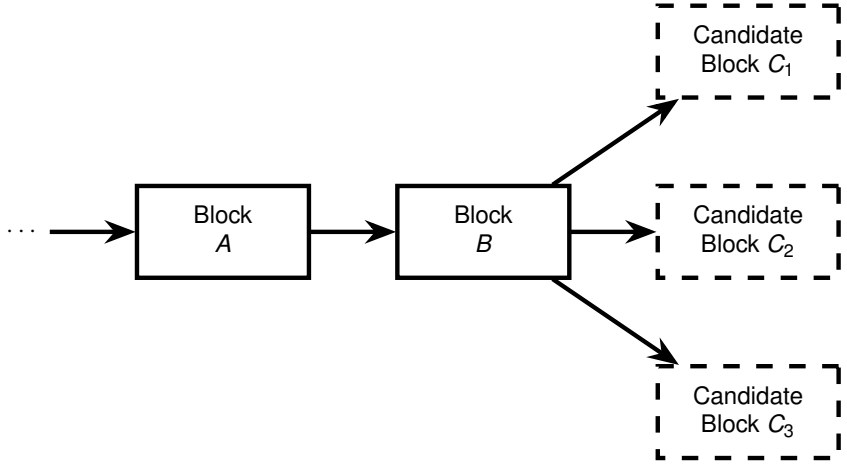
- Nodes broadcast transactions
- Miners accept valid transactions and reject invalid ones

Block Addition Workflow

- Nodes broadcast transactions
- Miners accept valid transactions and reject invalid ones (solves double spending)

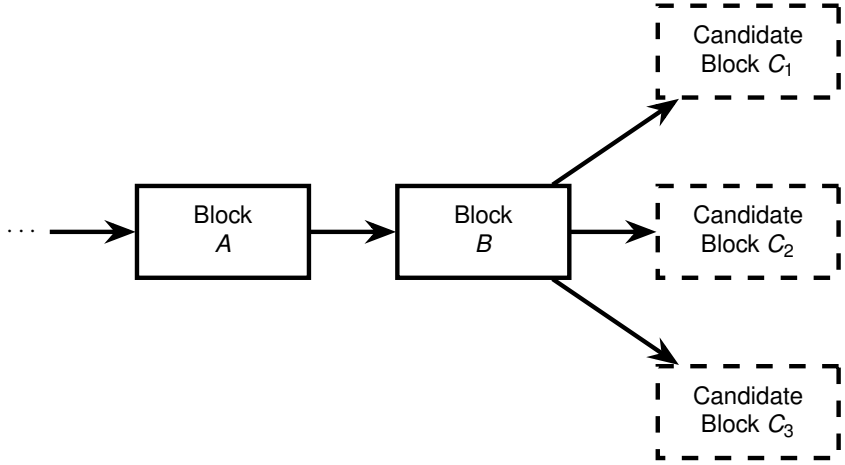
Block Addition Workflow

- Nodes broadcast transactions
- Miners accept valid transactions and reject invalid ones (solves double spending)
- Miners try extending the latest block



Block Addition Workflow

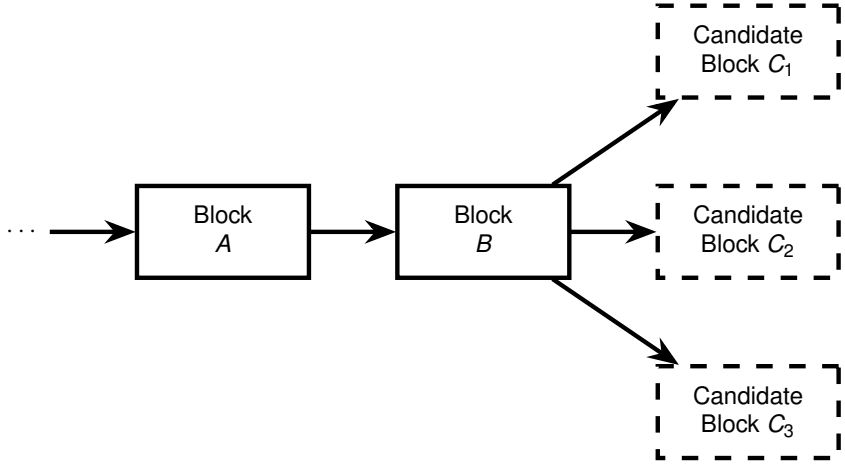
- Nodes broadcast transactions
- Miners accept valid transactions and reject invalid ones (solves double spending)
- Miners try extending the latest block



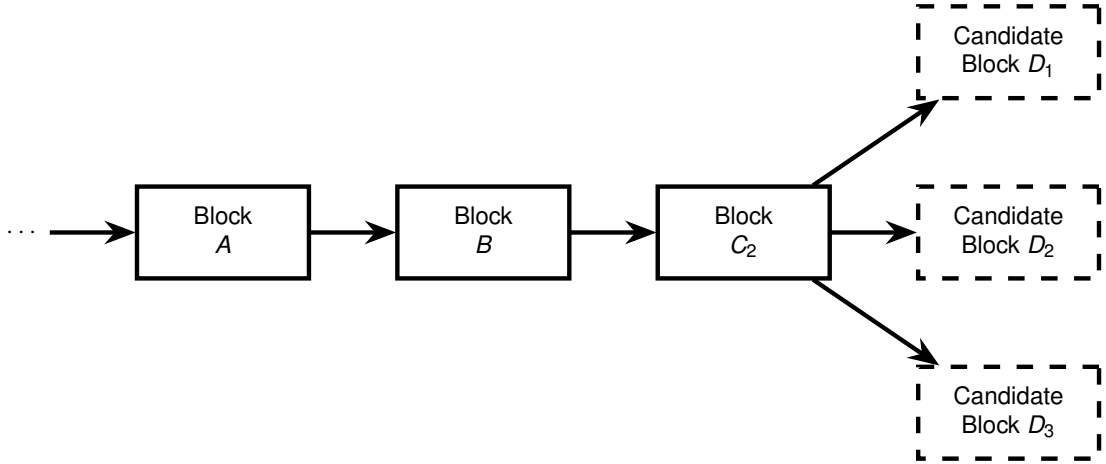
- Miners compete to solve the search puzzle and broadcast solutions

Block Addition Workflow

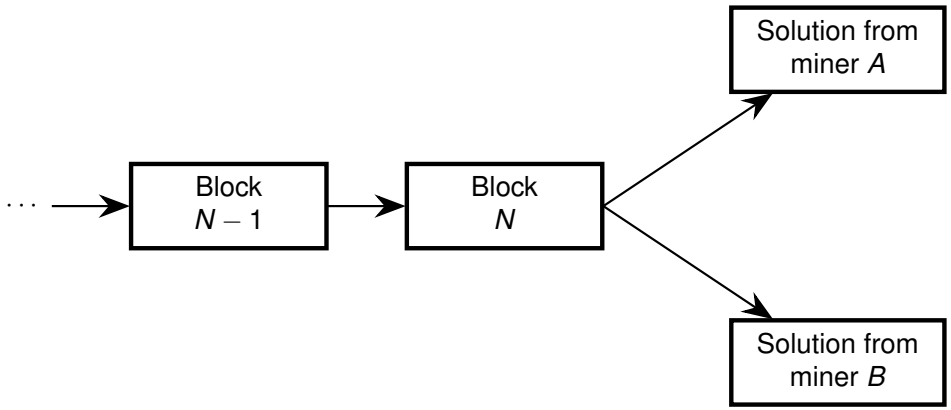
- Nodes broadcast transactions
- Miners accept valid transactions and reject invalid ones (solves double spending)
- Miners try extending the latest block



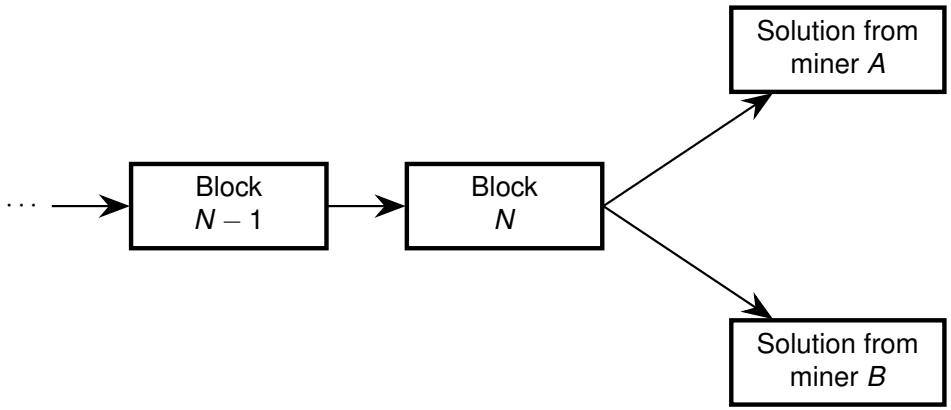
- Miners compete to solve the search puzzle and broadcast solutions
- Unsuccessful miners abandon their current candidate blocks and start work on new ones



What if two miners solve the puzzle at the same time?

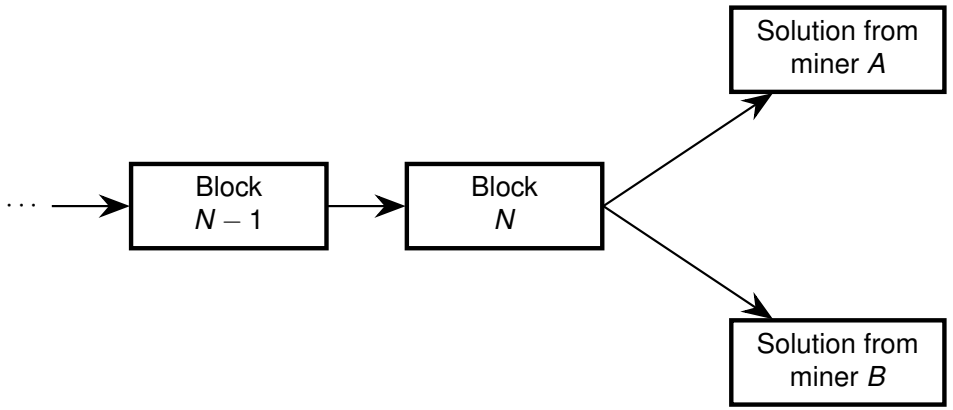


What if two miners solve the puzzle at the same time?



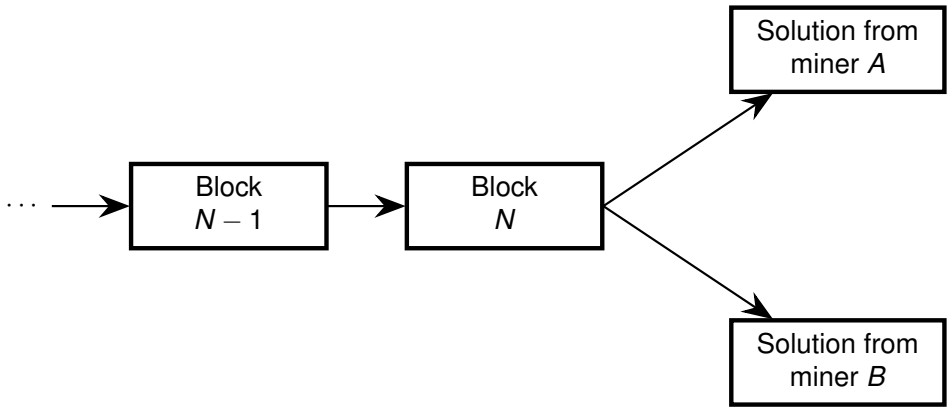
- Both miners will broadcast their solution on the network

What if two miners solve the puzzle at the same time?

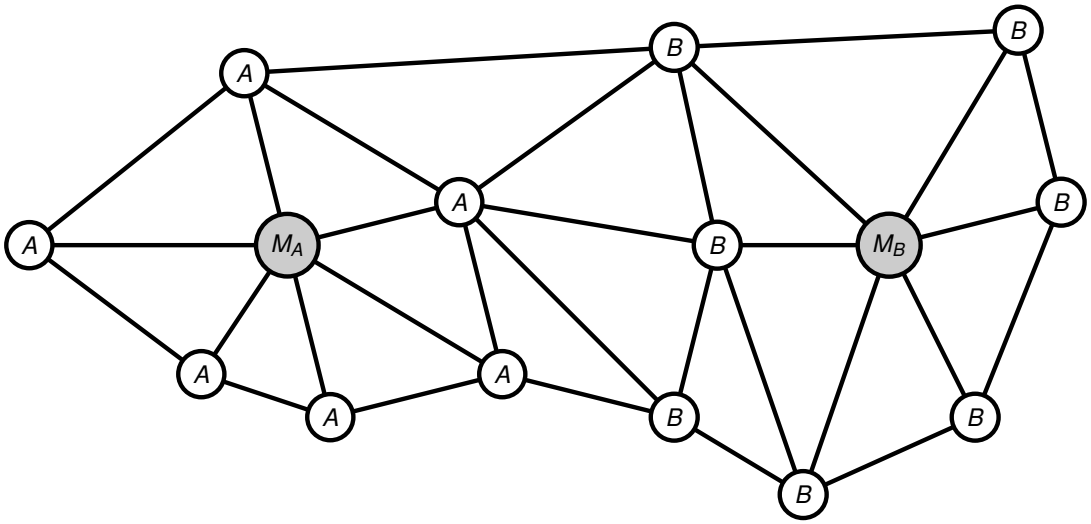


- Both miners will broadcast their solution on the network
- Nodes will accept the first solution they hear and reject others

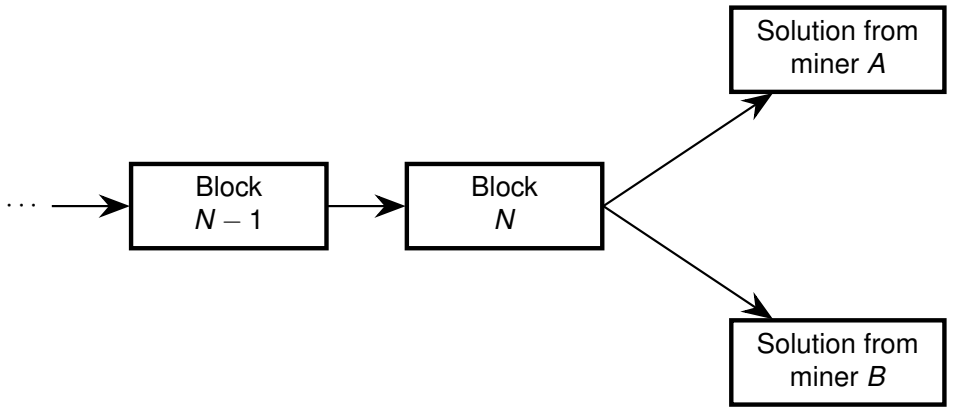
What if two miners solve the puzzle at the same time?



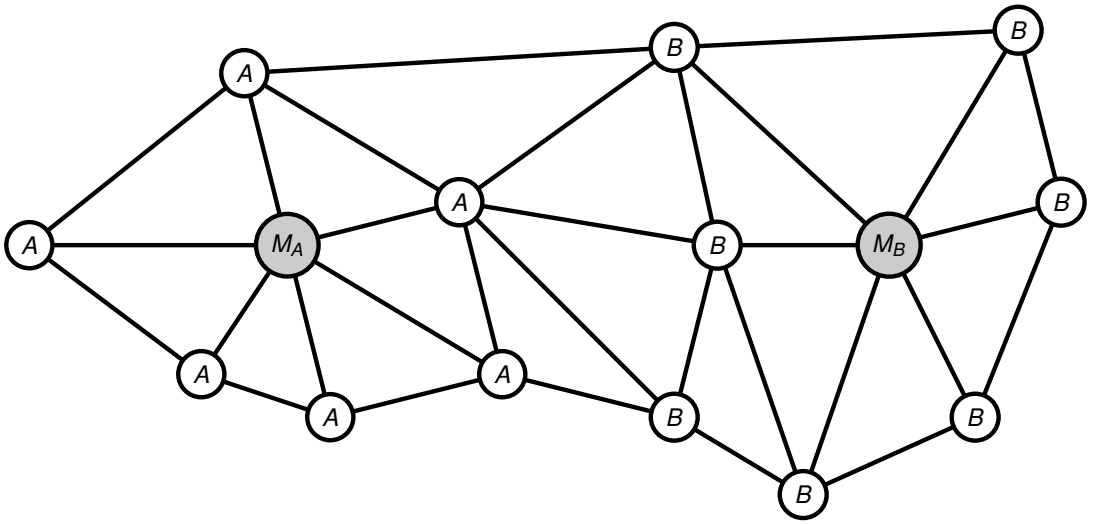
- Both miners will broadcast their solution on the network
- Nodes will accept the first solution they hear and reject others



What if two miners solve the puzzle at the same time?

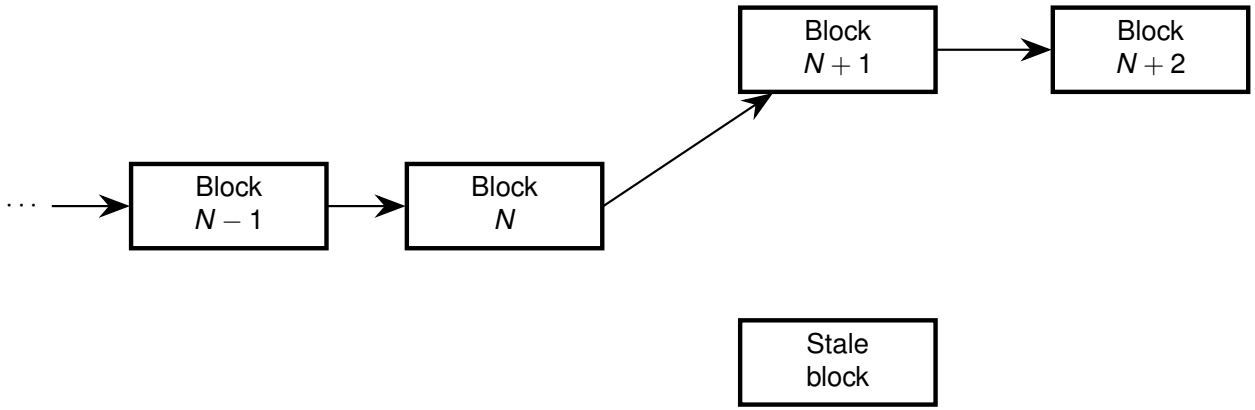


- Both miners will broadcast their solution on the network
- Nodes will accept the first solution they hear and reject others

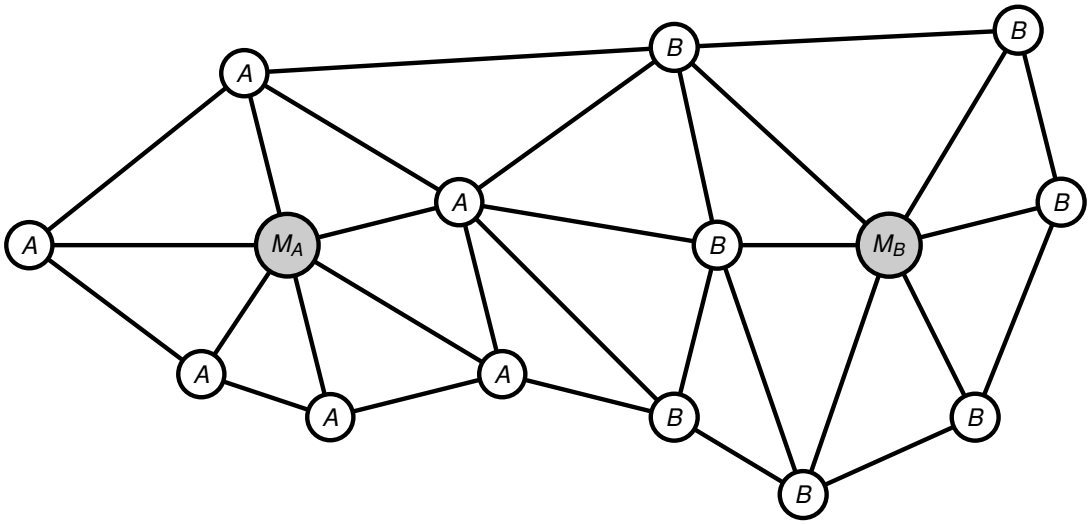


- Nodes always switch to the chain which was more difficult to produce

What if two miners solve the puzzle at the same time?

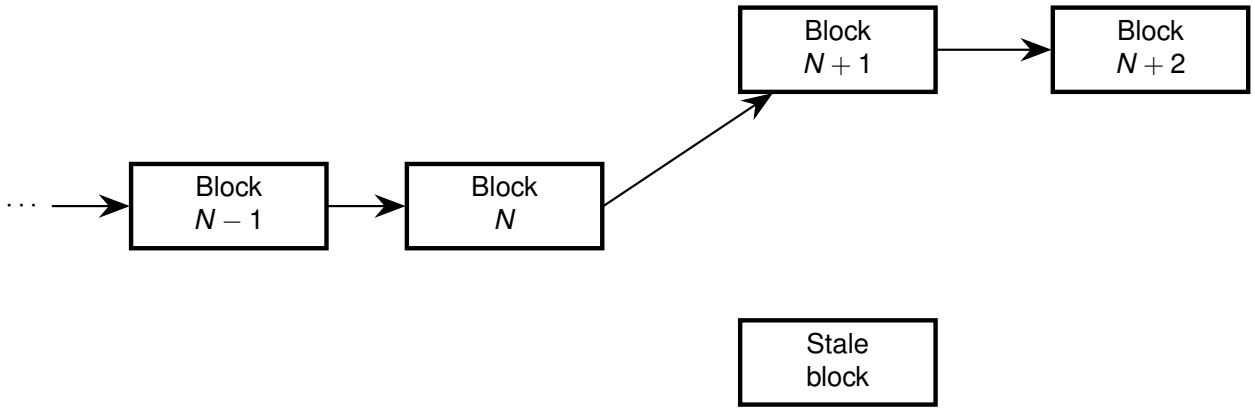


- Both miners will broadcast their solution on the network
- Nodes will accept the first solution they hear and reject others

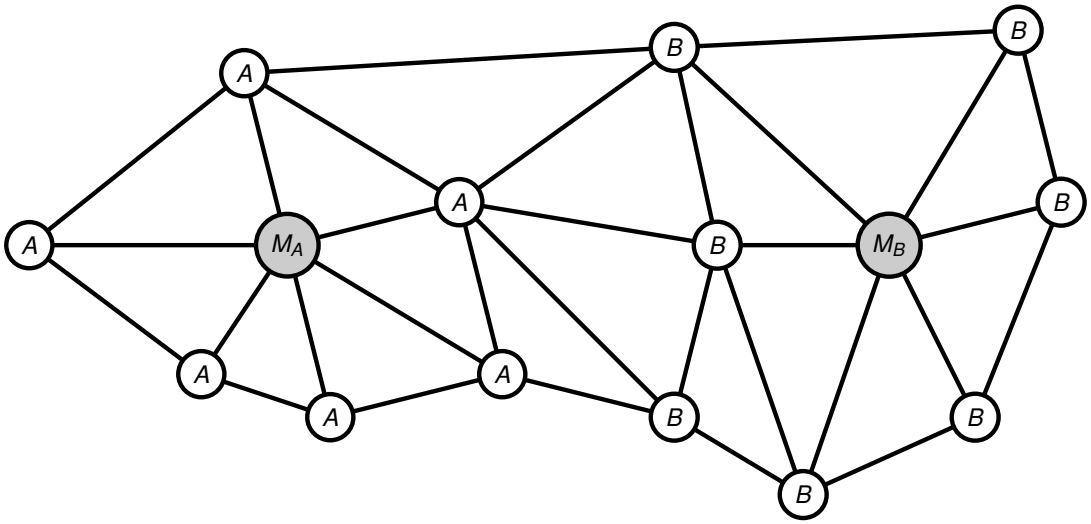


- Nodes always switch to the chain which was more difficult to produce

What if two miners solve the puzzle at the same time?

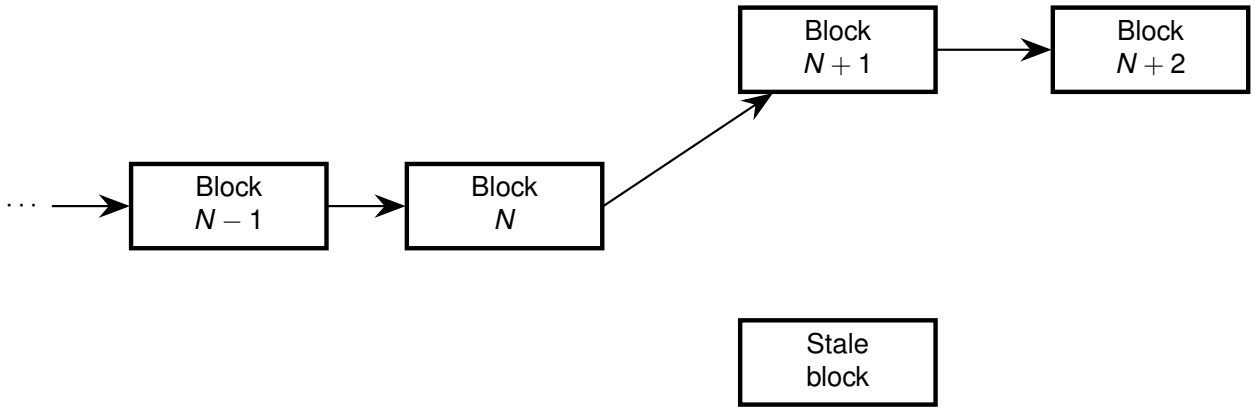


- Both miners will broadcast their solution on the network
- Nodes will accept the first solution they hear and reject others

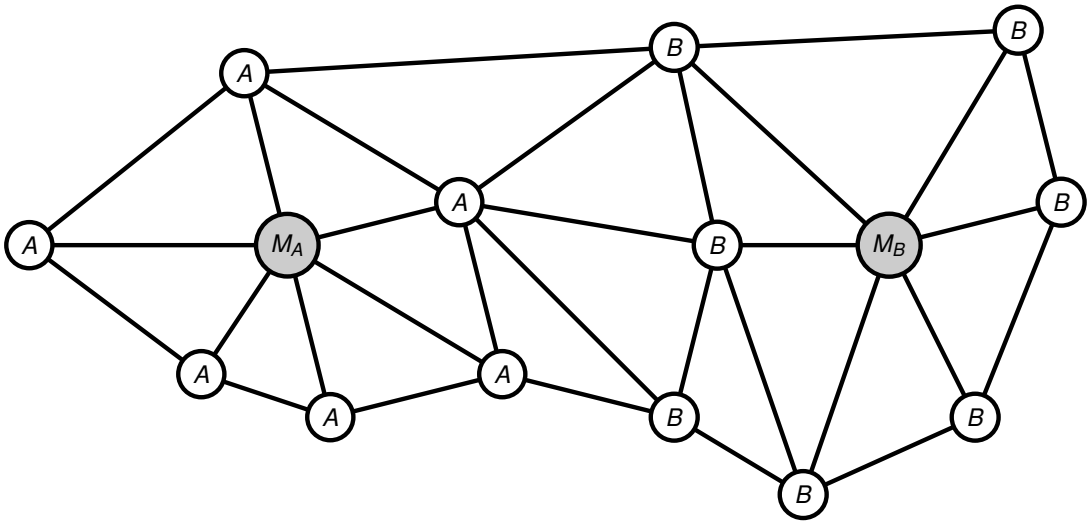


- Nodes always switch to the chain which was more difficult to produce
- Eventually the network will converge and achieve consensus

What if two miners solve the puzzle at the same time?



- Both miners will broadcast their solution on the network
- Nodes will accept the first solution they hear and reject others



- Nodes always switch to the chain which was more difficult to produce
- Eventually the network will converge and achieve consensus
- This is called proof-of-work (PoW) consensus

How often are new blocks created?

How often are new blocks created?

- Once every 10 minutes

How often are new blocks created?

- Once every 10 minutes

nVersion
hashPrevBlock
hashMerkleRoot
nTime
nBits
nNonce

How often are new blocks created?

- Once every 10 minutes

nVersion
hashPrevBlock
hashMerkleRoot
nTime
nBits
nNonce

- Every 2016 blocks, the target T is recalculated

How often are new blocks created?

- Once every 10 minutes

nVersion
hashPrevBlock
hashMerkleRoot
nTime
nBits
nNonce

- Every 2016 blocks, the target T is recalculated
- Let t_{sum} = Number of seconds taken to mine last 2016 blocks

How often are new blocks created?

- Once every 10 minutes

nVersion
hashPrevBlock
hashMerkleRoot
nTime
nBits
nNonce

- Every 2016 blocks, the target T is recalculated
- Let t_{sum} = Number of seconds taken to mine last 2016 blocks

$$T_{\text{new}} = \frac{t_{\text{sum}}}{2016 \times 10 \times 60} \times T$$

How often are new blocks created?

- Once every 10 minutes

nVersion
hashPrevBlock
hashMerkleRoot
nTime
nBits
nNonce

- Every 2016 blocks, the target T is recalculated
- Let t_{sum} = Number of seconds taken to mine last 2016 blocks

$$T_{\text{new}} = \frac{t_{\text{sum}}}{2016 \times 10 \times 60} \times T$$

- Recall that probability of success in single trial is $\frac{T+1}{2^{256}}$

How often are new blocks created?

- Once every 10 minutes

nVersion
hashPrevBlock
hashMerkleRoot
nTime
nBits
nNonce

- Every 2016 blocks, the target T is recalculated
- Let t_{sum} = Number of seconds taken to mine last 2016 blocks

$$T_{\text{new}} = \frac{t_{\text{sum}}}{2016 \times 10 \times 60} \times T$$

- Recall that probability of success in single trial is $\frac{T+1}{2^{256}}$
- If $t_{\text{sum}} = 2016 \times 8 \times 60$, then $T_{\text{new}} = \frac{4}{5} T$

How often are new blocks created?

- Once every 10 minutes

nVersion
hashPrevBlock
hashMerkleRoot
nTime
nBits
nNonce

- Every 2016 blocks, the target T is recalculated
- Let t_{sum} = Number of seconds taken to mine last 2016 blocks

$$T_{\text{new}} = \frac{t_{\text{sum}}}{2016 \times 10 \times 60} \times T$$

- Recall that probability of success in single trial is $\frac{T+1}{2^{256}}$
- If $t_{\text{sum}} = 2016 \times 8 \times 60$, then $T_{\text{new}} = \frac{4}{5} T$
- If $t_{\text{sum}} = 2016 \times 12 \times 60$, then $T_{\text{new}} = \frac{6}{5} T$

Bitcoin Blockchain Explorers

Bitcoin Blockchain Explorers

- Web interfaces to view current blockchain state

Bitcoin Blockchain Explorers

- Web interfaces to view current blockchain state
  - `https://www.blockstream.info`

Bitcoin Blockchain Explorers

- Web interfaces to view current blockchain state
  - `https://www.blockstream.info`
  - `https://www.blockchain.com/explorer`

Bitcoin Blockchain Explorers

- Web interfaces to view current blockchain state
  - `https://www.blockstream.info`
  - `https://www.blockchain.com/explorer`
- Demo checklist

Bitcoin Blockchain Explorers

- Web interfaces to view current blockchain state
  - `https://www.blockstream.info`
  - `https://www.blockchain.com/explorer`
- Demo checklist
  - List of transactions (coinbase, regular)

Bitcoin Blockchain Explorers

- Web interfaces to view current blockchain state
  - `https://www.blockstream.info`
  - `https://www.blockchain.com/explorer`
- Demo checklist
  - List of transactions (coinbase, regular)
  - Address generation in `https://www.bitaddress.org`

Bitcoin Blockchain Explorers

- Web interfaces to view current blockchain state
  - `https://www.blockstream.info`
  - `https://www.blockchain.com/explorer`
- Demo checklist
  - List of transactions (coinbase, regular)
  - Address generation in `https://www.bitaddress.org`
  - Brainwallet generation at `https://brainwalletx.github.io`

Bitcoin Supply

Bitcoin Supply

- The block subsidy was initially 50 BTC per block

Bitcoin Supply

- The block subsidy was initially 50 BTC per block
- Halves every 210,000 blocks

Bitcoin Supply

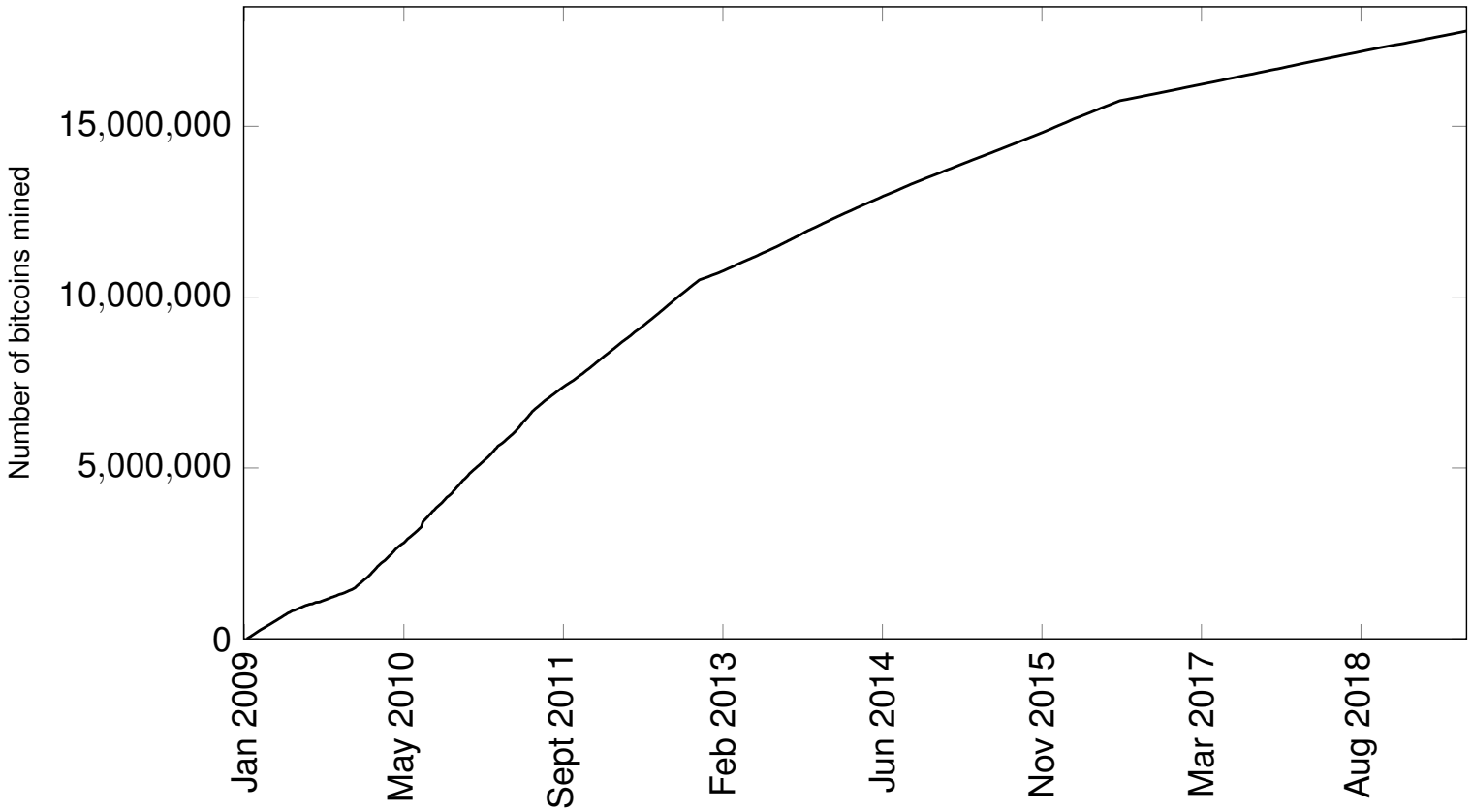
- The block subsidy was initially 50 BTC per block
- Halves every 210,000 blocks $\approx$ 4 years

Bitcoin Supply

- The block subsidy was initially 50 BTC per block
- Halves every 210,000 blocks $\approx$ 4 years
- Became 25 BTC in Nov 2012 and 12.5 BTC in July 2016

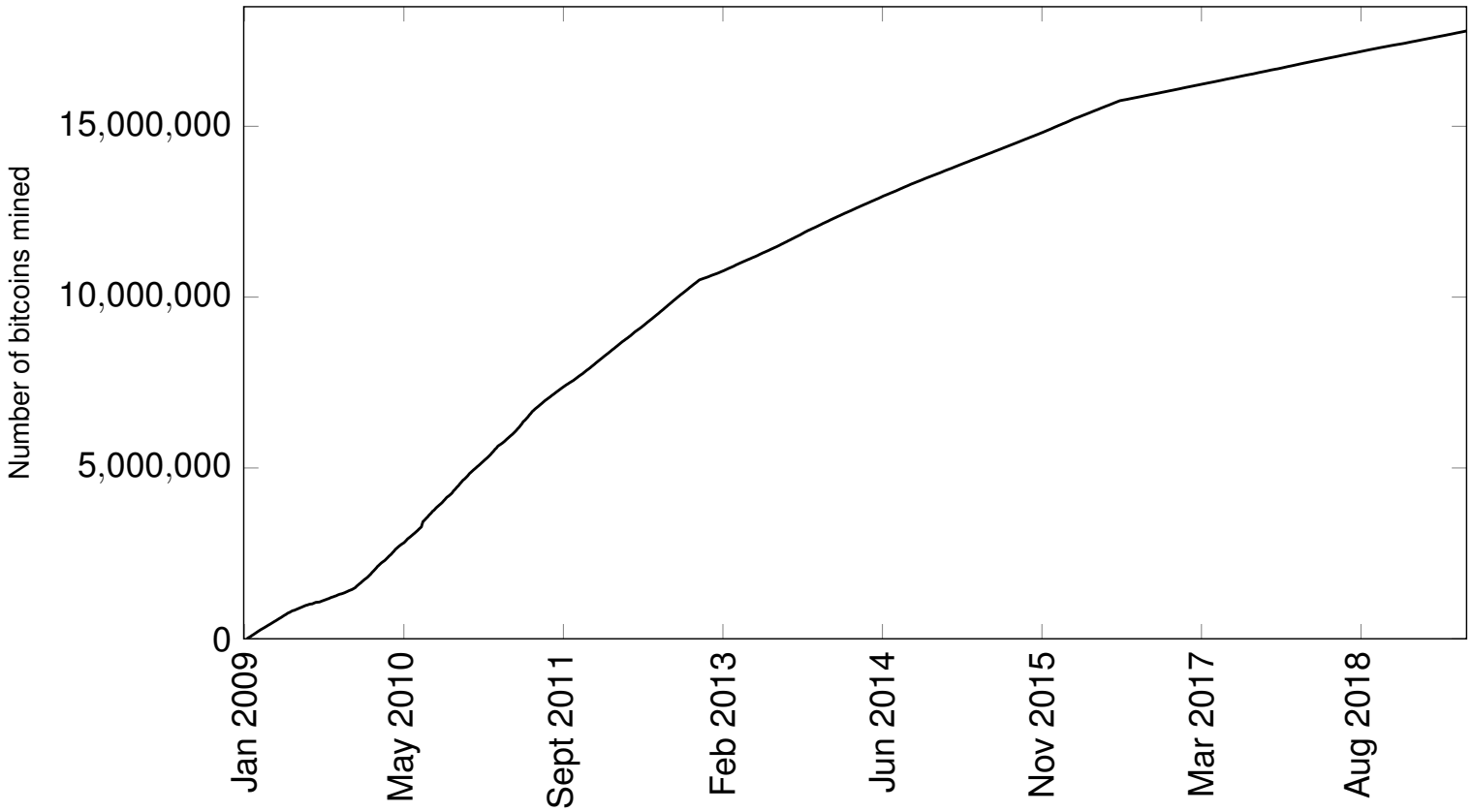
Bitcoin Supply

- The block subsidy was initially 50 BTC per block
- Halves every 210,000 blocks $\approx$ 4 years
- Became 25 BTC in Nov 2012 and 12.5 BTC in July 2016
- Total Bitcoin supply is 21 million



Bitcoin Supply

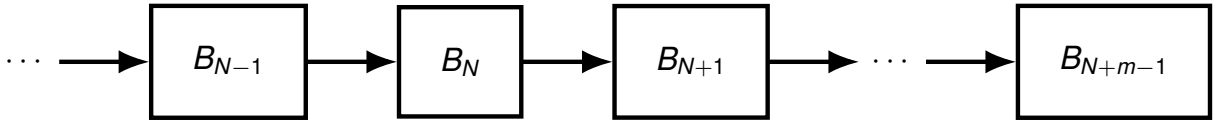
- The block subsidy was initially 50 BTC per block
- Halves every 210,000 blocks $\approx$ 4 years
- Became 25 BTC in Nov 2012 and 12.5 BTC in July 2016
- Total Bitcoin supply is 21 million



- The last bitcoin will be mined in 2140

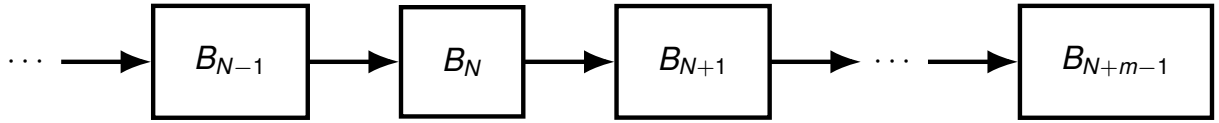
Tamper Resistance

- Suppose Alice wants to modify block B_N

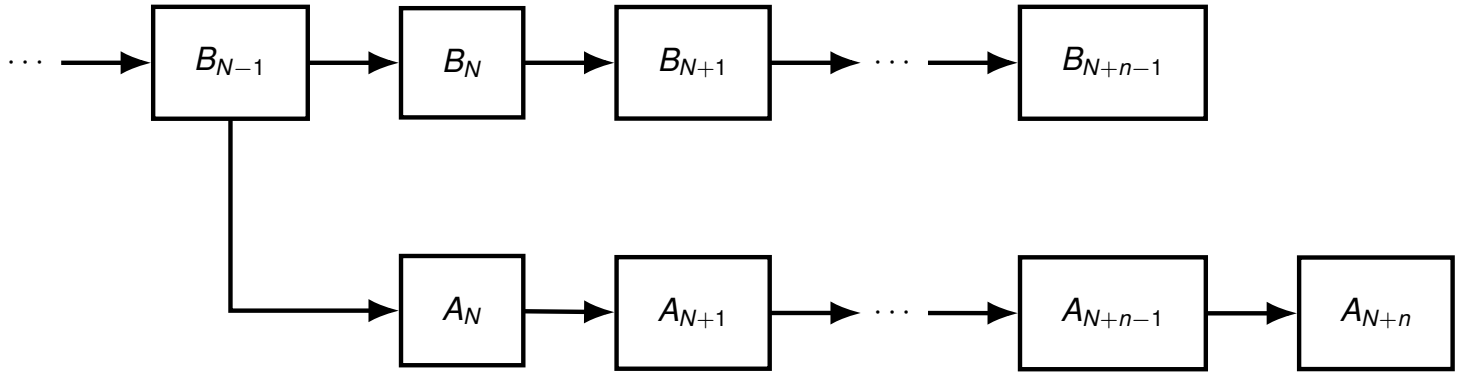


Tamper Resistance

- Suppose Alice wants to modify block B_N

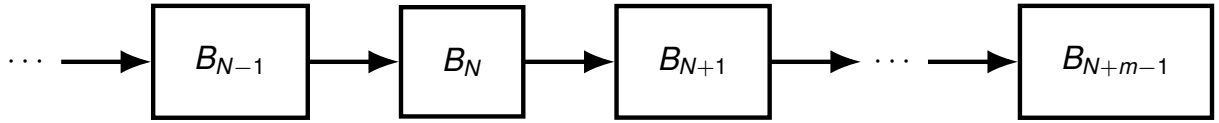


- Alice works on A_N branch; other miners work on B_N branch

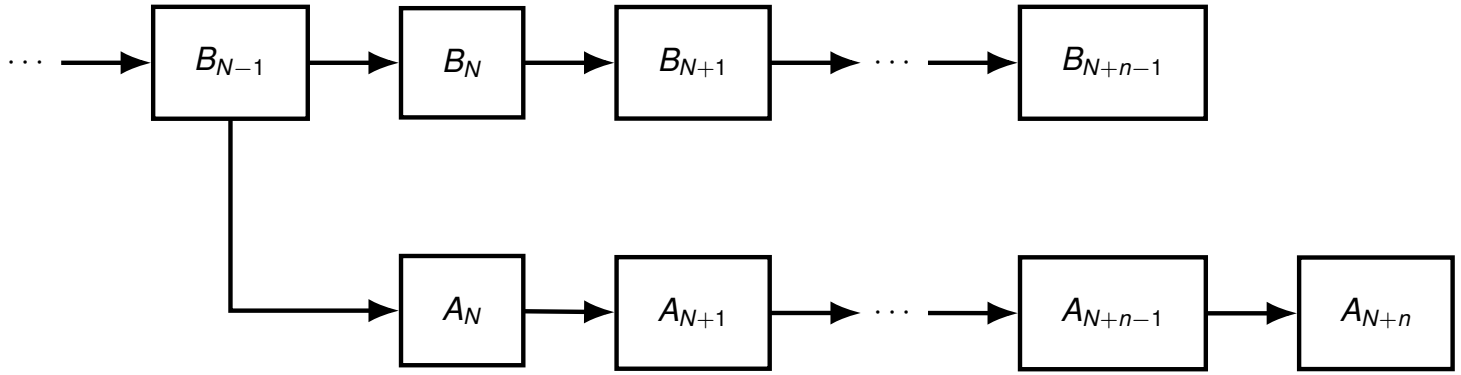


Tamper Resistance

- Suppose Alice wants to modify block B_N



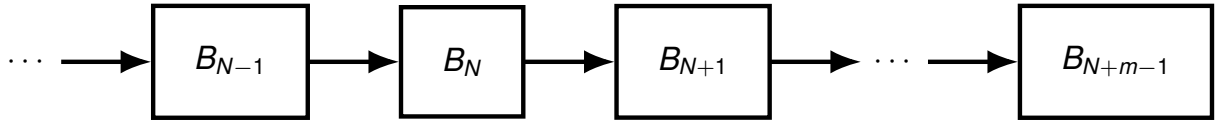
- Alice works on A_N branch; other miners work on B_N branch



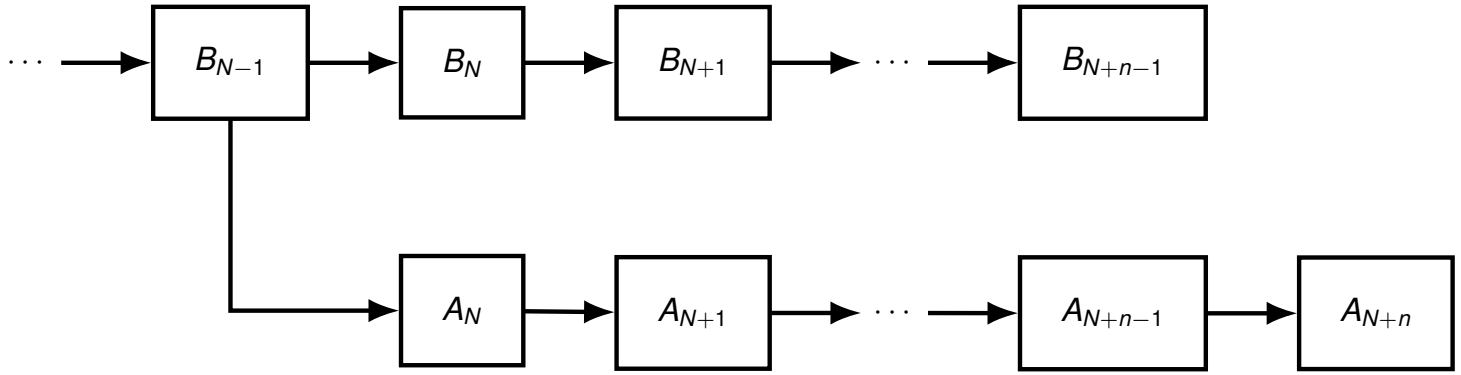
- She needs to mine blocks faster than the rest of the miners

Tamper Resistance

- Suppose Alice wants to modify block B_N



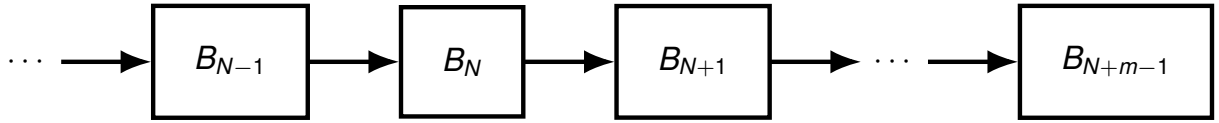
- Alice works on A_N branch; other miners work on B_N branch



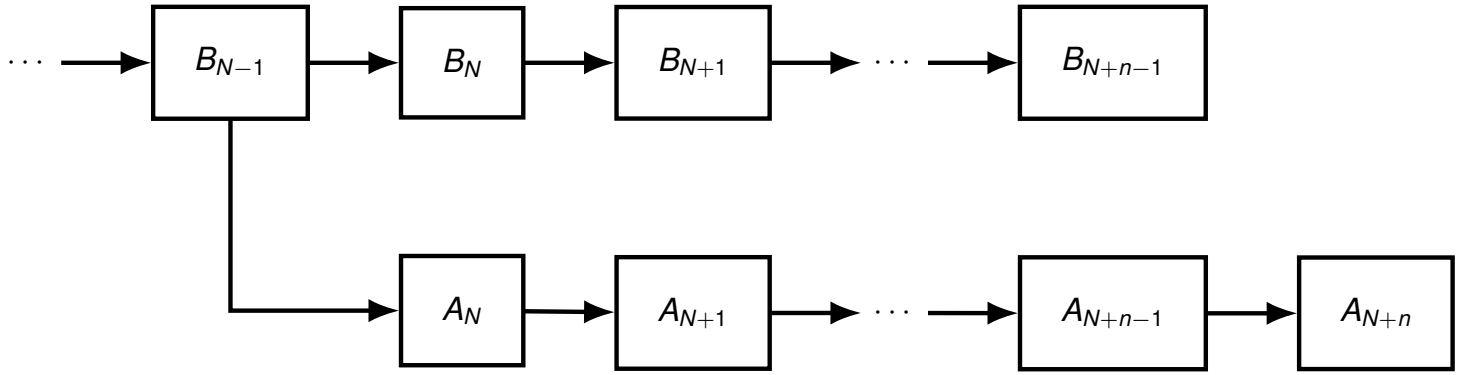
- She needs to mine blocks faster than the rest of the miners
- Possible if she controls 50% or more of network hashrate

Tamper Resistance

- Suppose Alice wants to modify block B_N



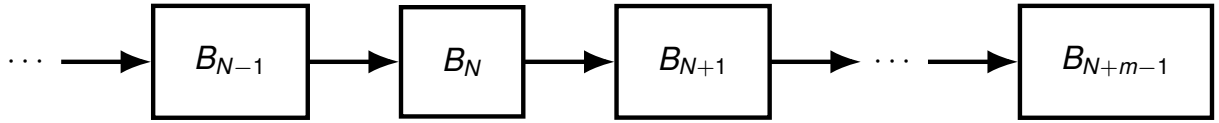
- Alice works on A_N branch; other miners work on B_N branch



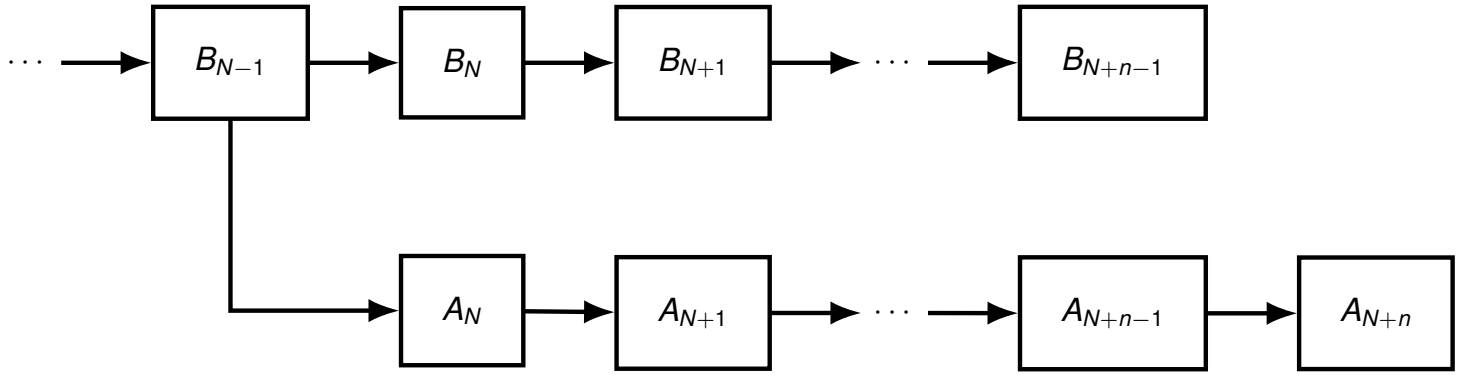
- She needs to mine blocks faster than the rest of the miners
- Possible if she controls 50% or more of network hashrate
- Current Bitcoin network hashrate $\approx$ 78 EH/s = 78×10^{18} H/s

Tamper Resistance

- Suppose Alice wants to modify block B_N



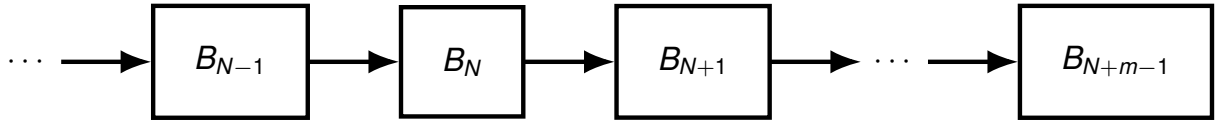
- Alice works on A_N branch; other miners work on B_N branch



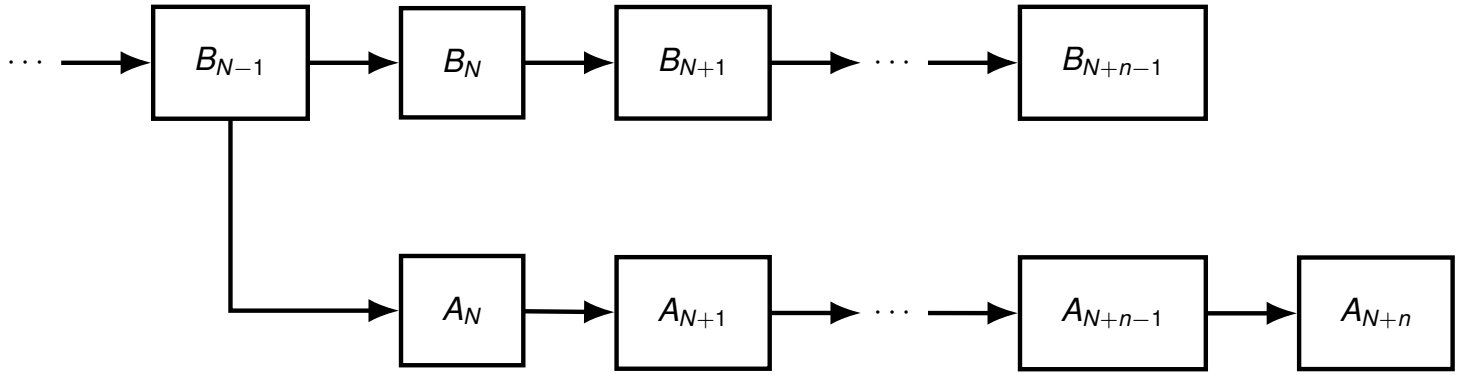
- She needs to mine blocks faster than the rest of the miners
- Possible if she controls 50% or more of network hashrate
- Current Bitcoin network hashrate $\approx$ 78 EH/s = 78×10^{18} H/s
- One mining unit costing \$350 gives 16 TH/s

Tamper Resistance

- Suppose Alice wants to modify block B_N

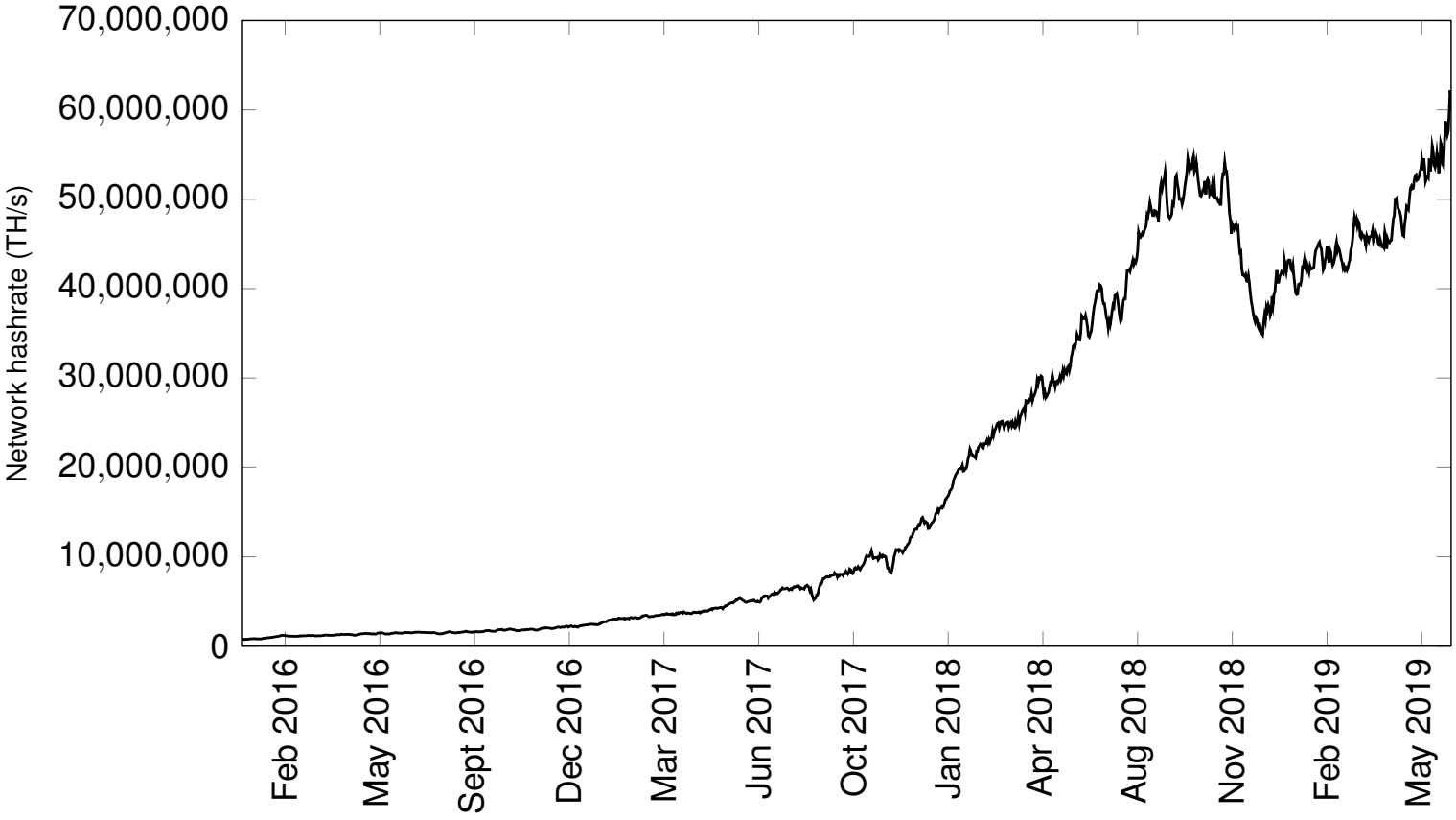


- Alice works on A_N branch; other miners work on B_N branch



- She needs to mine blocks faster than the rest of the miners
- Possible if she controls 50% or more of network hashrate
- Current Bitcoin network hashrate $\approx$ 78 EH/s = 78×10^{18} H/s
- One mining unit costing \$350 gives 16 TH/s
- Controlling 50% of hashrate = Controlling 853 million USD worth of hardware

Bitcoin Hashrate



Data source: `https://www.blockchain.com/charts/hash-rate`

Key Takeaways

Key Takeaways

- Bitcoin's blockchain prevents double spending and tampering

Key Takeaways

- Bitcoin's blockchain prevents double spending and tampering
- Secure only if nobody controls 50% or more of network hashrate

Key Takeaways

- Bitcoin's blockchain prevents double spending and tampering
- Secure only if nobody controls 50% or more of network hashrate
- Mining difficulty adjusted to regulate coin supply

Key Takeaways

- Bitcoin's blockchain prevents double spending and tampering
- Secure only if nobody controls 50% or more of network hashrate
- Mining difficulty adjusted to regulate coin supply
- Miners incentivized by block reward

Key Takeaways

- Bitcoin's blockchain prevents double spending and tampering
- Secure only if nobody controls 50% or more of network hashrate
- Mining difficulty adjusted to regulate coin supply
- Miners incentivized by block reward
- Block subsidy halves every four years to cap total coin supply

Blockstream Satellite

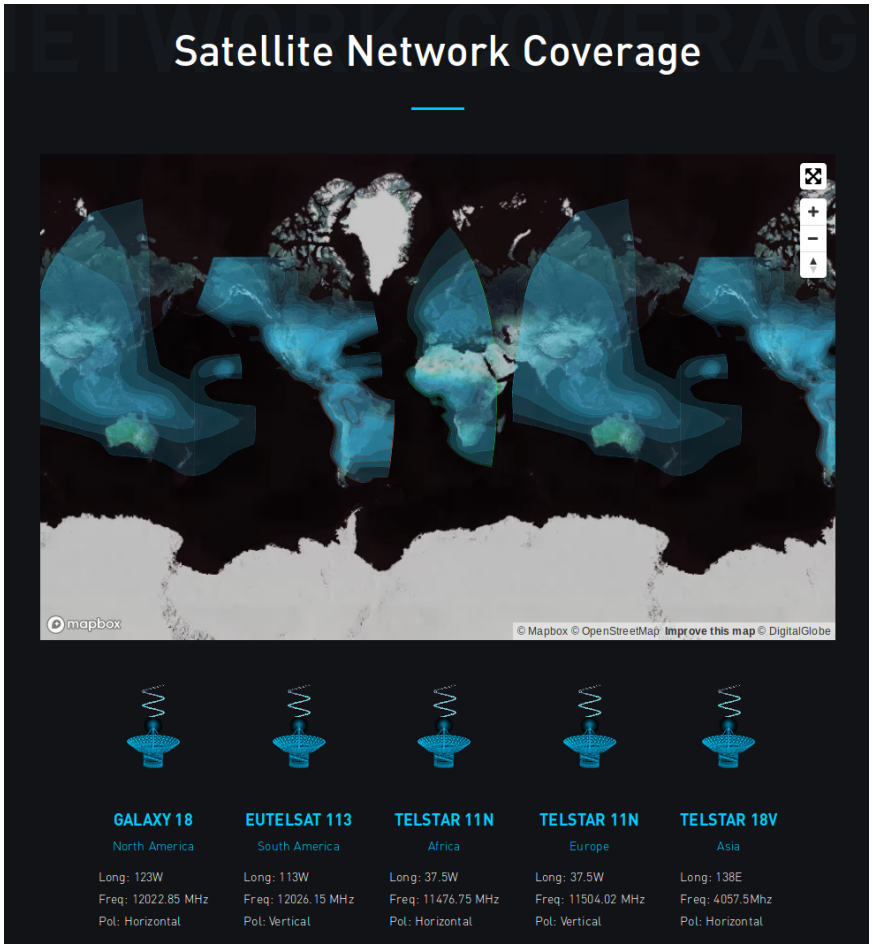


Image credit: `https://blockstream.com/satellite/`

- Blockstream Satellite network broadcasts the Bitcoin blockchain for free
- No Internet required to receive blocks (verify payments in Bitcoin)

How Blockstream Satellite Works?

Image credit: `https://blockstream.com/satellite/`

- Ground stations (teleports) participate in the Bitcoin network and transmit blocks to geosynchronous satellites
- Satellites receive the blocks and broadcast them across the Earth
- Anyone in the coverage area with a small satellite antenna and an inexpensive USB receiver can receive these blocks
- Anyone can verify large payments in remote areas

Bitcoin Testnet Transactions

Bitcoin Testnet Transactions

- Each cryptocurrency has a mainnet and one or more testnets

Bitcoin Testnet Transactions

- Each cryptocurrency has a mainnet and one or more testnets
- Bitcoin Testnet
  `https://live.blockcypher.com/btc-testnet/`

Bitcoin Testnet Transactions

- Each cryptocurrency has a mainnet and one or more testnets
- Bitcoin Testnet `https://live.blockcypher.com/btc-testnet/`
- Testnet Address Generator `https://bitcoinpaperwallet.com/bitcoinpaperwallet/generate-wallet.html?design=alt-testnet`

Bitcoin Testnet Transactions

- Each cryptocurrency has a mainnet and one or more testnets
- Bitcoin Testnet `https://live.blockcypher.com/btc-testnet/`
- Testnet Address Generator `https://bitcoinpaperwallet.com/bitcoinpaperwallet/generate-wallet.html?design=alt-testnet`
- Testnet faucet 1 `https://coinfaucet.eu/en/btc-testnet/`

Bitcoin Testnet Transactions

- Each cryptocurrency has a mainnet and one or more testnets
- Bitcoin Testnet `https://live.blockcypher.com/btc-testnet/`
- Testnet Address Generator `https://bitcoinpaperwallet.com/bitcoinpaperwallet/generate-wallet.html?design=alt-testnet`
- Testnet faucet 1 `https://coinfaucet.eu/en/btc-testnet/`
- Testnet faucet 2 `https://bitcoinfaucet.uo1.net`

Bitcoin Testnet Transactions

- Each cryptocurrency has a mainnet and one or more testnets
- Bitcoin Testnet `https://live.blockcypher.com/btc-testnet/`
- Testnet Address Generator `https://bitcoinpaperwallet.com/bitcoinpaperwallet/generate-wallet.html?design=alt-testnet`
- Testnet faucet 1 `https://coinfaucet.eu/en/btc-testnet/`
- Testnet faucet 2 `https://bitcoinfaucet.uo1.net`
- Mycelium Testnet Wallet Mobile App

References

- Chapter 4 of *An Introduction to Bitcoin*, S. Vijayakumaran, `www.ee.iitb.ac.in/~sarva/bitcoin.html`
- Bitcoin Charts `https://www.blockchain.com/charts`
- Bitmain Mining Rigs `https://shop.bitmain.com`
- Blockstream Satellite `https://blockstream.com/satellite/`