MIT Digital Currency Initiative and the University of Brasilia presents

# Cryptocurrency Design and Engineering

Lecture 12: Scalability
Taught by: Neha Narula
October 21st, 2025
MAS.S62

# First half of today

- Motivation: transaction throughput
- What is scalability?
- On-chain scaling and challenges
- Payment channels
- The Lightning Network

# Motivation: transaction throughput

| System | Throughput |
|---|---|
| Bitcoin | 7 txs/sec |
| Ethereum | 15-30 txs/sec |
| Solana | 3,200 txs/sec |
| PostgreSQL `r5.4xlarge` | 5,700 txs/sec |
| Visa | 24,000 txs/sec |

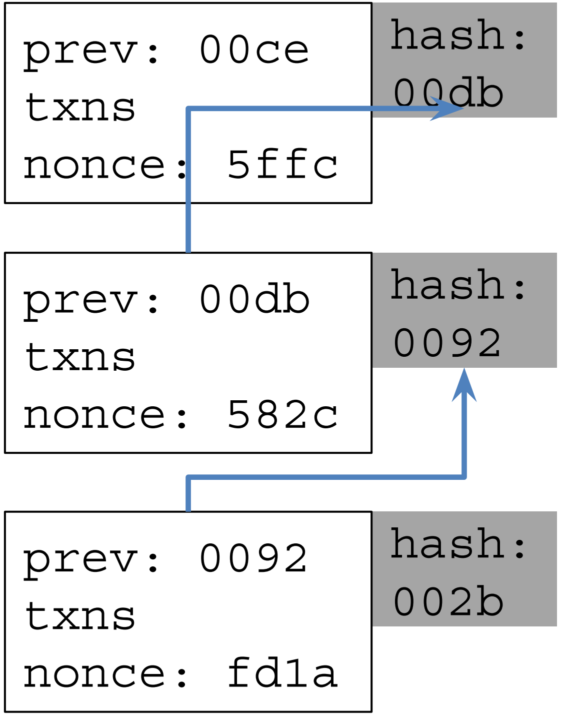
tokenterminal.com/explorer/projects/solana/metrics/transactions-per-second

docs.aws.amazon.com/whitepapers/latest/optimizing-postgresql-on-ec2-using-ebs/postgresql-benchmark-observations-and-considerations.html

# What is scalability?

- More transactions per second?
- Cheaper fees?
- Lower latency?
- All of the above?

Scalability is when you can add more resources to get more good work done



prev: 00ce
txns
nonce: 5ffc

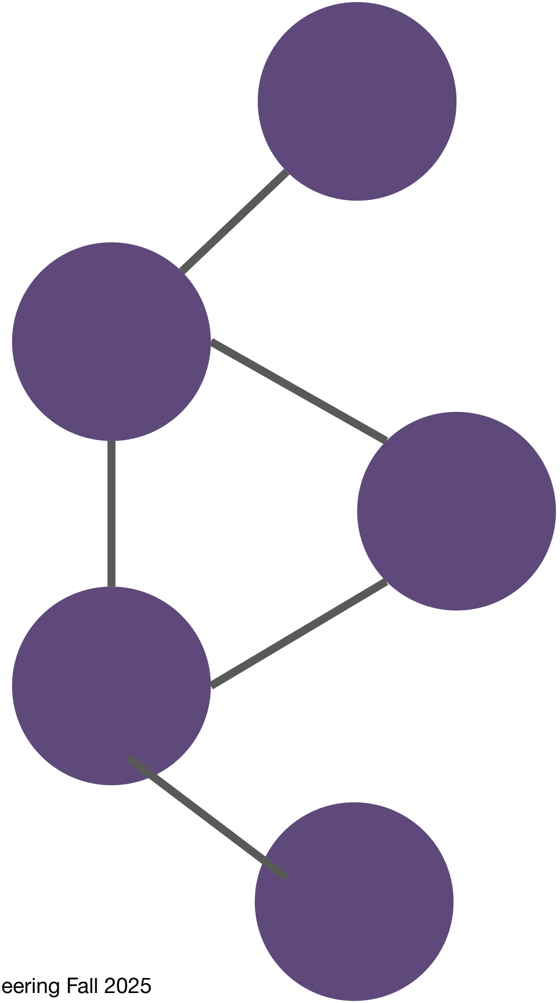
hash: 00db

prev: 00db
txns
nonce: 582c

hash: 0092

prev: 0092
txns
nonce: fd1a

hash: 002b



Every new node still has to do the same work (validating the chain), again

More resources don't lead to additional good work

# Ways to add more resources

- Vertical scaling
  - Get a more powerful computer
- Horizontal scaling
  - Add more {cores, computers} and parallelize the work

# Vertical scaling

- 4MB blocks every 10 minutes: hardcoded. Why? Could we just change these limits?
  - Block size wars
  - Ethereum: gas limit, 12 sec block time
- Tension with decentralization
  - More expensive to run a fully validating node
  - Longer to validate leads to more orphans which leads to mining centralization

Market incentives lecture on November 18th!

# Horizontal scaling techniques

- Sharding: not everyone validates everything
- Verifiable computation: prover / verifiers
- Batched verification: payment channels

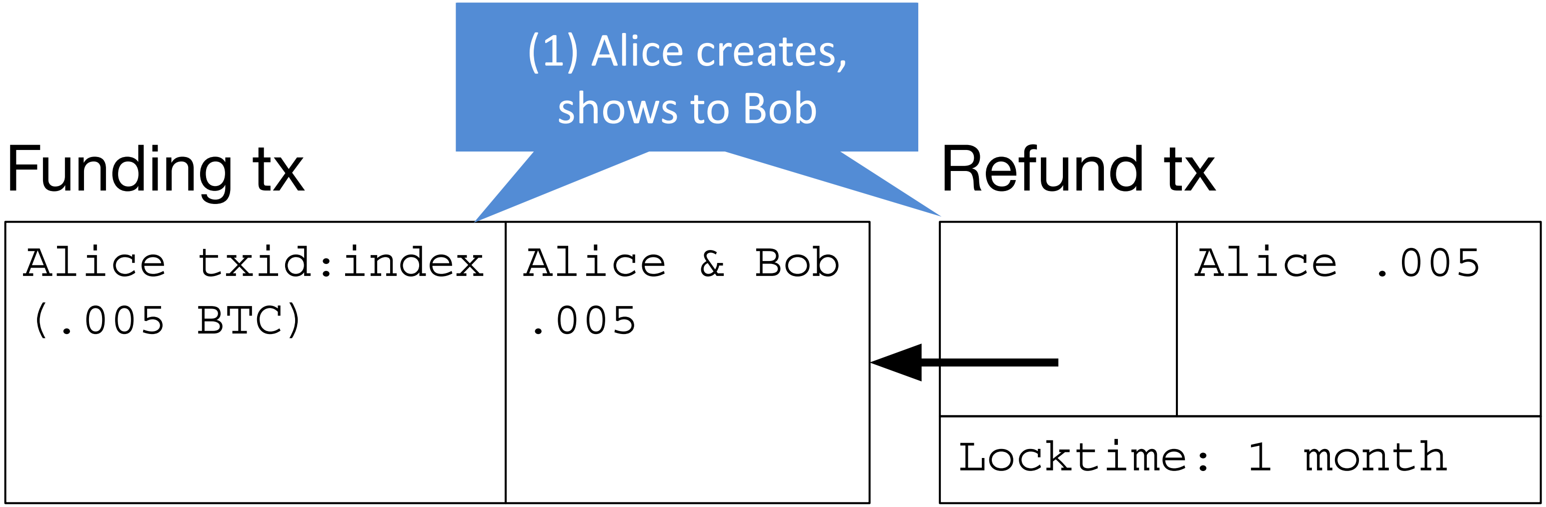
# First half of today

- Motivation: transaction throughput
- What is scalability?
- On-chain scaling and challenges
- **Payment channels**
- The Lightning Network

# Payment channels: basic idea

- Alice repeatedly buys coffee from Bob, who runs a coffeeshop
- Alice can run up a bar tab
  - Alice deposits .005 BTC in the tab
  - Every day, Alice buys coffee
  - At the end of the month, settle the tab between Alice and Bob
- Collapses 30 payments □ 2 on-chain transactions!

# Channel setup



(1) Alice creates, shows to Bob

Funding tx

Alice txid:index (.005 BTC)

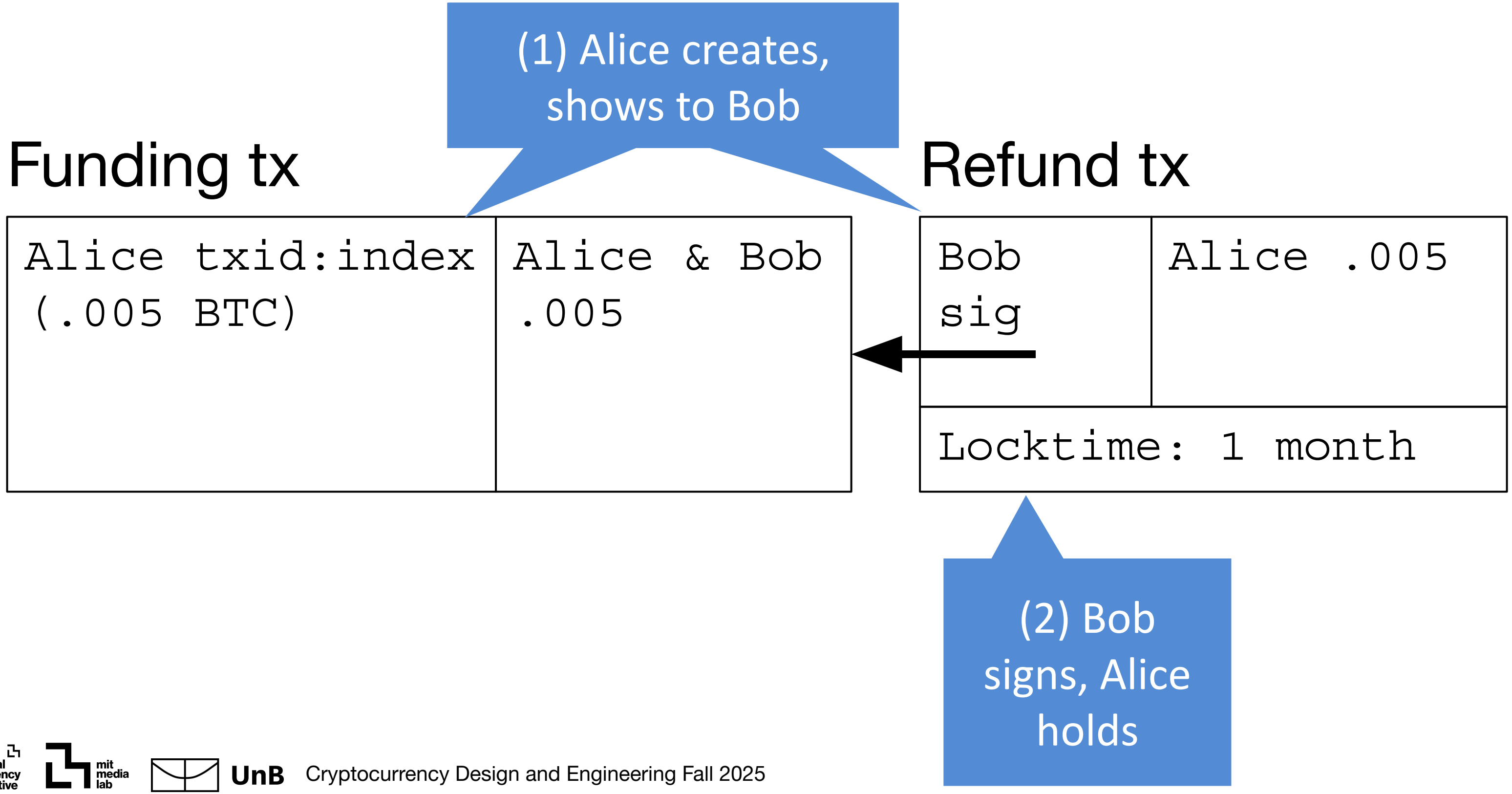
Alice & Bob .005

Refund tx

Alice .005

Locktime: 1 month

# Channel setup



(1) Alice creates, shows to Bob

Funding tx

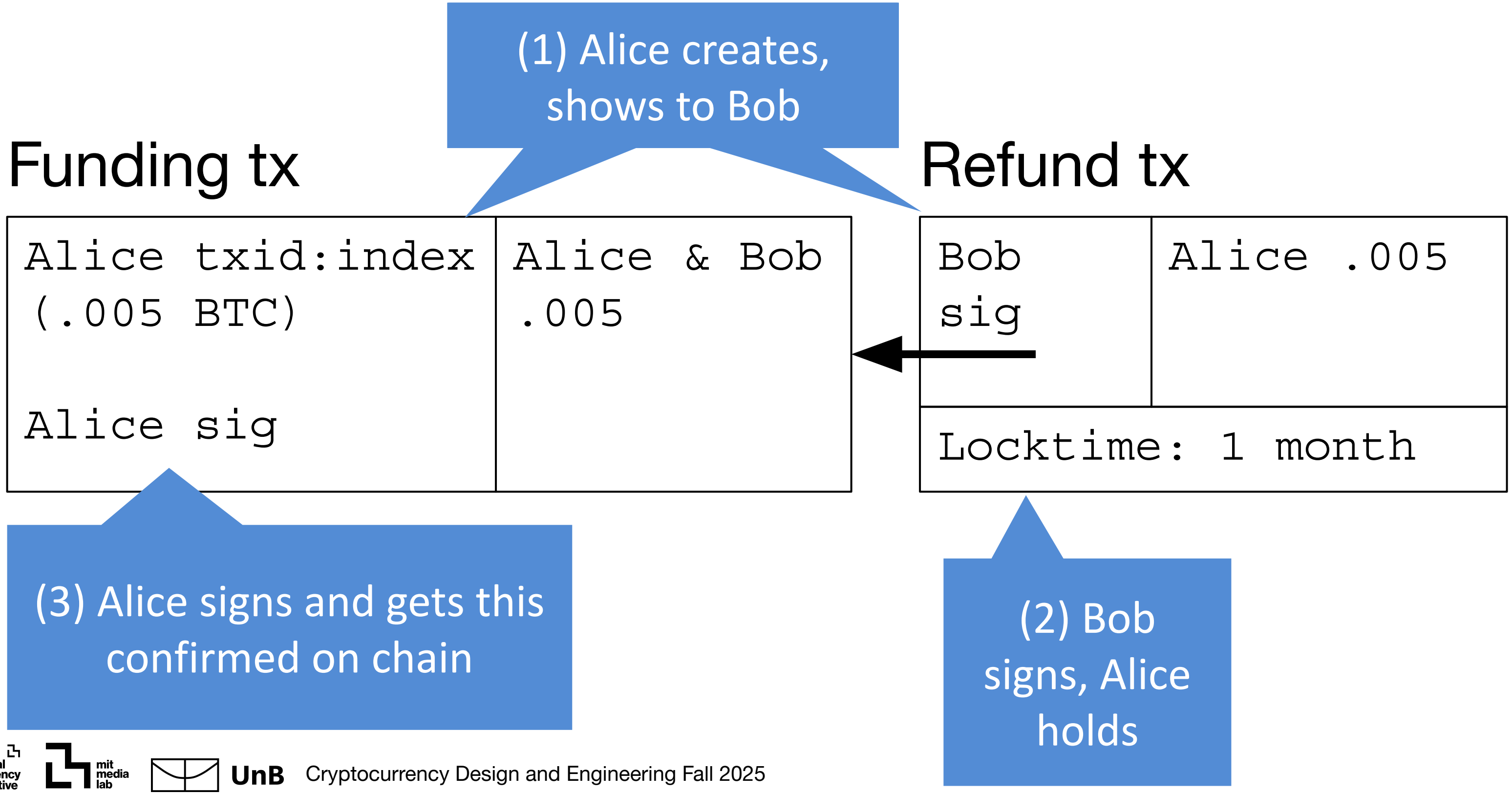
| | |
|---|---|
| Alice txid:index (.005 BTC) | Alice & Bob .005 |

Refund tx

| | |
|---|---|
| Bob sig | Alice .005 |
| Locktime: 1 month | |

(2) Bob signs, Alice holds

# Channel setup



(1) Alice creates, shows to Bob

Funding tx

| | |
|---|---|
| Alice txid:index<br>(.005 BTC)<br><br>Alice sig | Alice & Bob<br>.005 |

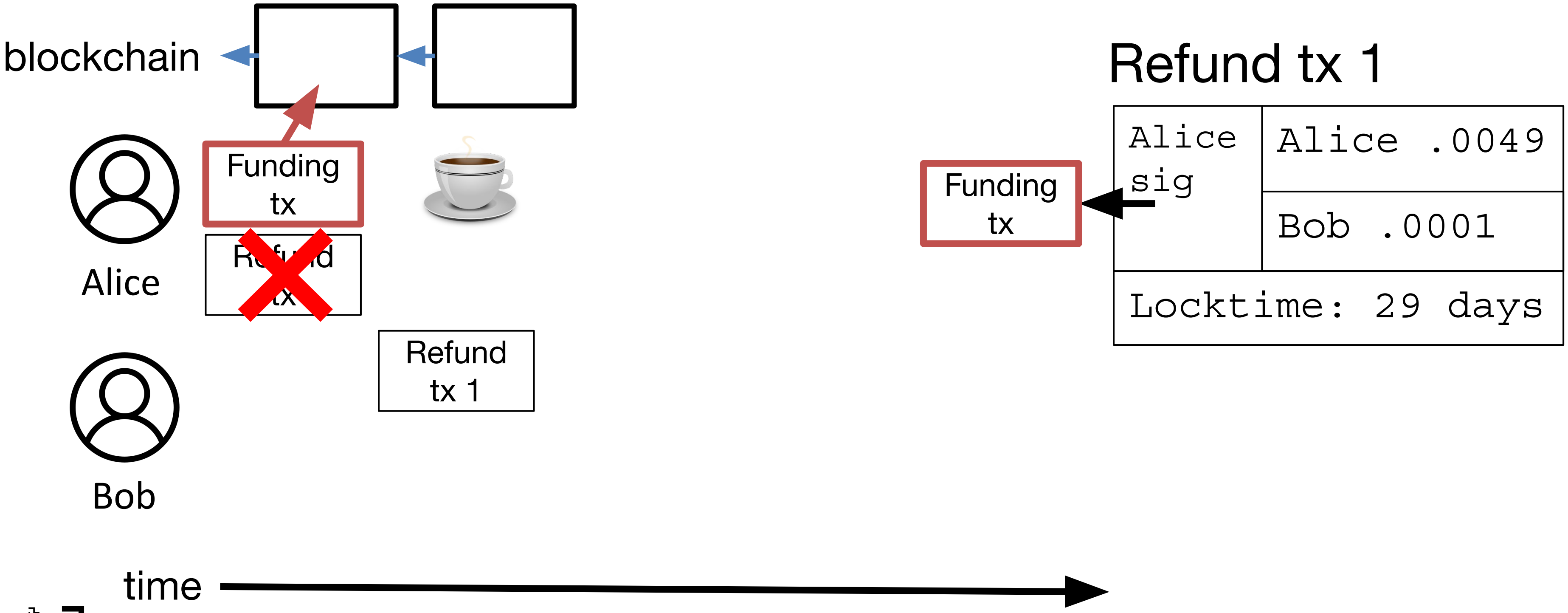
Refund tx

| | |
|---|---|
| Bob<br>sig | Alice .005 |
| Locktime: 1 month | |

(3) Alice signs and gets this confirmed on chain

(2) Bob signs, Alice holds

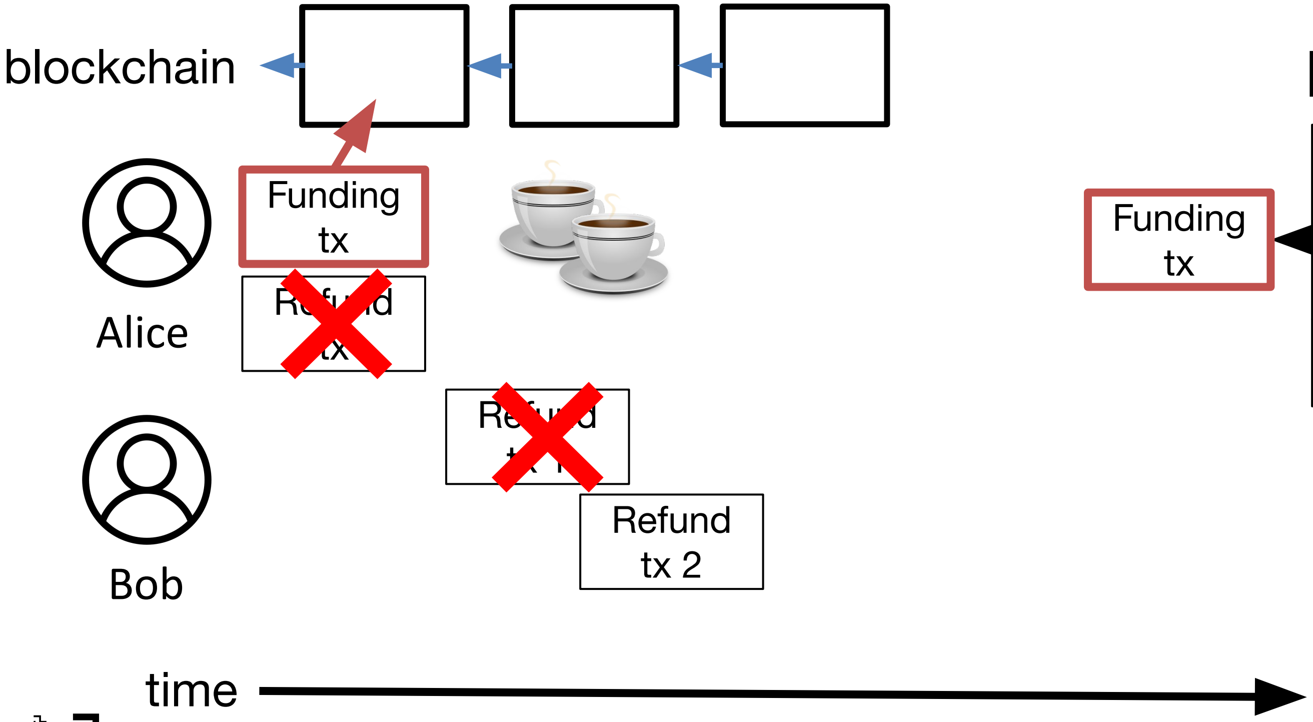
# Many payments Alice --> Bob



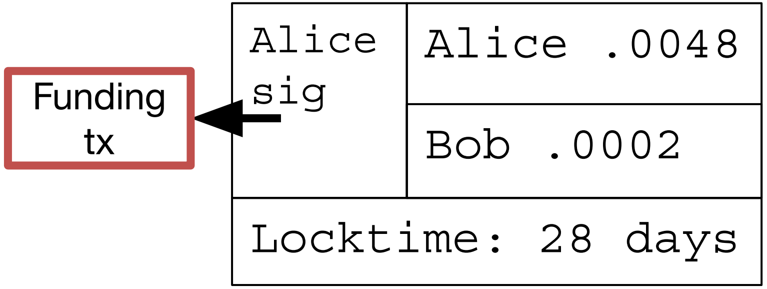
## Refund tx 1

| | |
|---|---|
| Alice sig | Alice .0049 |
| | Bob .0001 |
| Locktime: 29 days | |

# Many payments Alice --> Bob

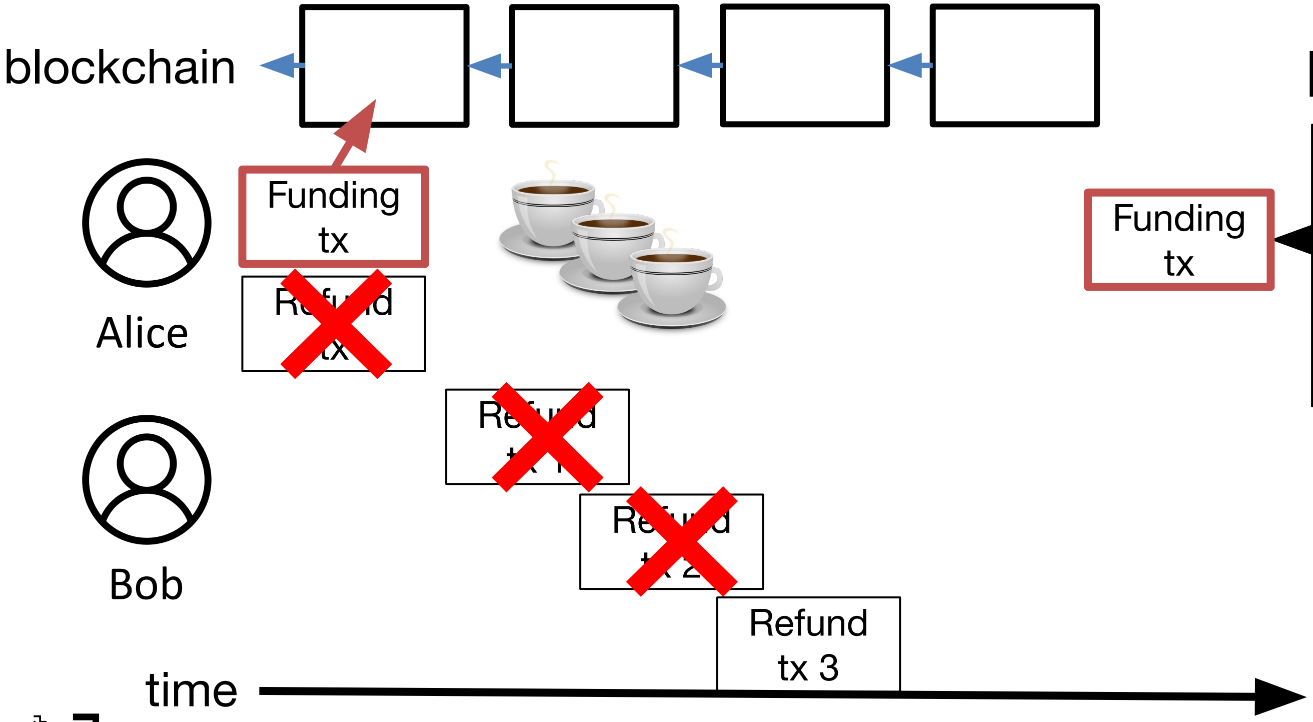


## Refund tx 2



|  |  |
|---|---|
| Alice sig | Alice .0048<br>Bob .0002 |
| Locktime: 28 days |  |

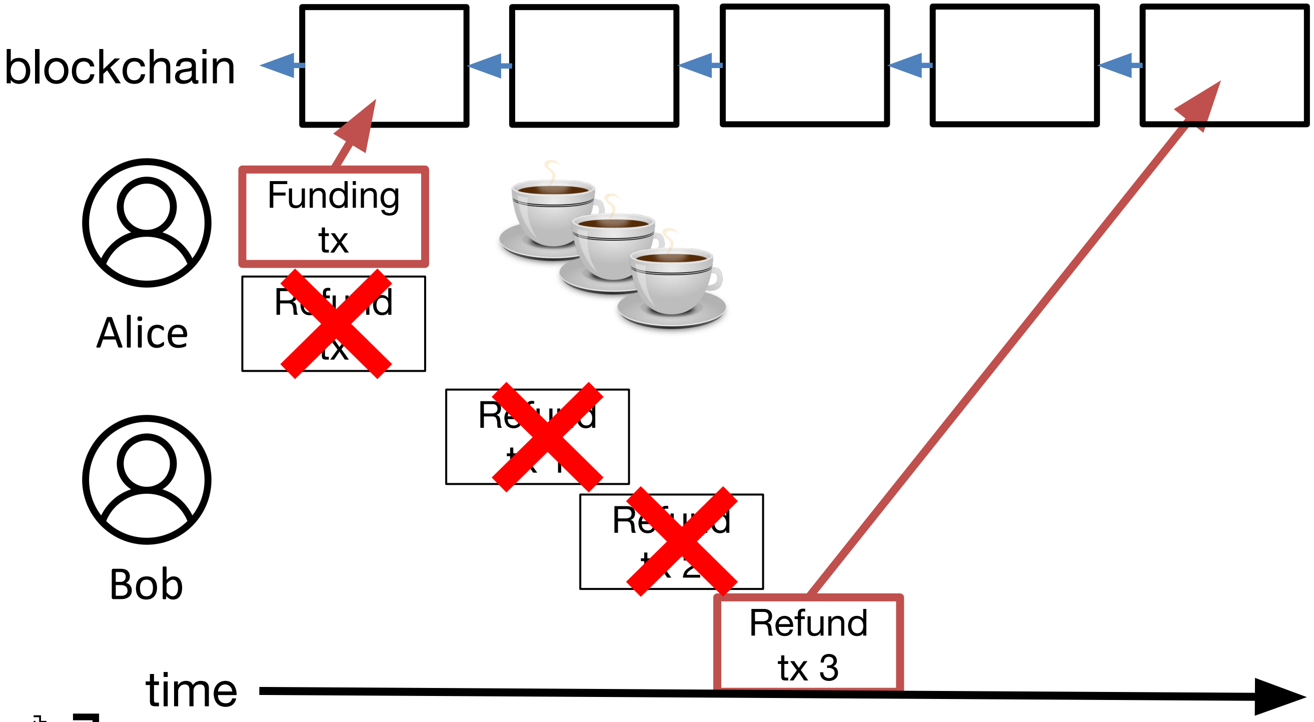
# Many payments Alice --> Bob



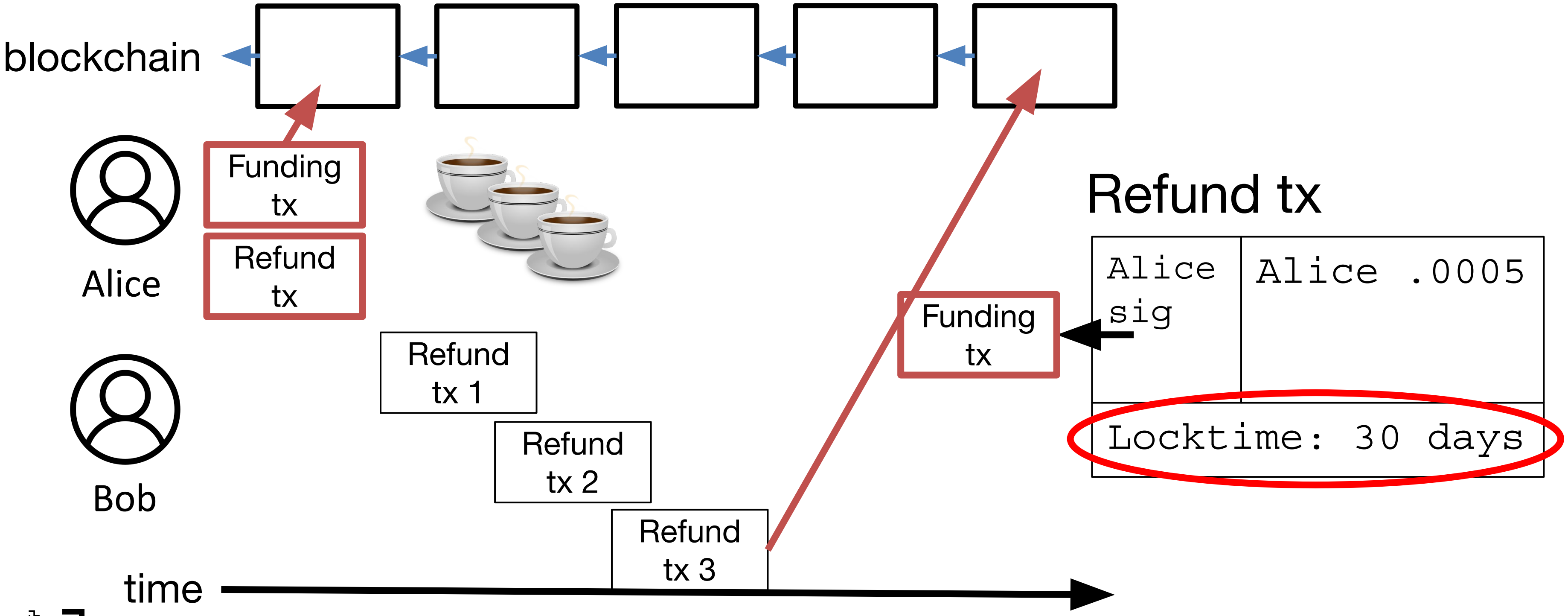
## Refund tx 3

|  |  |
|---|---|
| Alice sig | Alice .0047 |
|  | Bob .0003 |
| Locktime: 27 days |  |

# Bob closes the payment channel



# Alice maliciously closes the channel



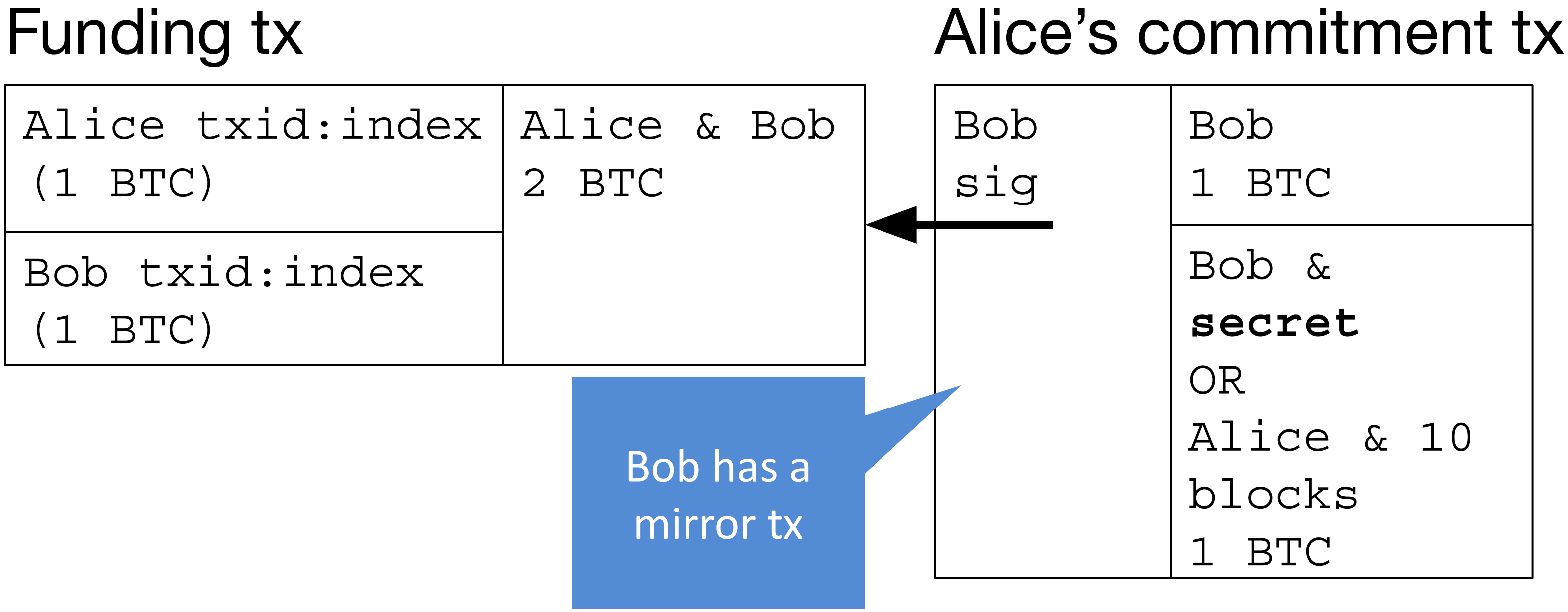
Refund tx

| | |
|---|---|
| Alice sig | Alice .0005 |
| Locktime: 30 days | |

# Bidirectional channels

- Previous payment channels only allowed one-way payments (Alice to Bob)
- Can we do two-way?
  - Yes!
- Similar idea. Differences:
  - Both sides fund the channel
  - Revoke old states explicitly

# Bidirectional channel setup



Funding tx

| | |
|---|---|
| Alice txid:index<br>(1 BTC) | Alice & Bob<br>2 BTC |
| Bob txid:index<br>(1 BTC) | |

Alice’s commitment tx

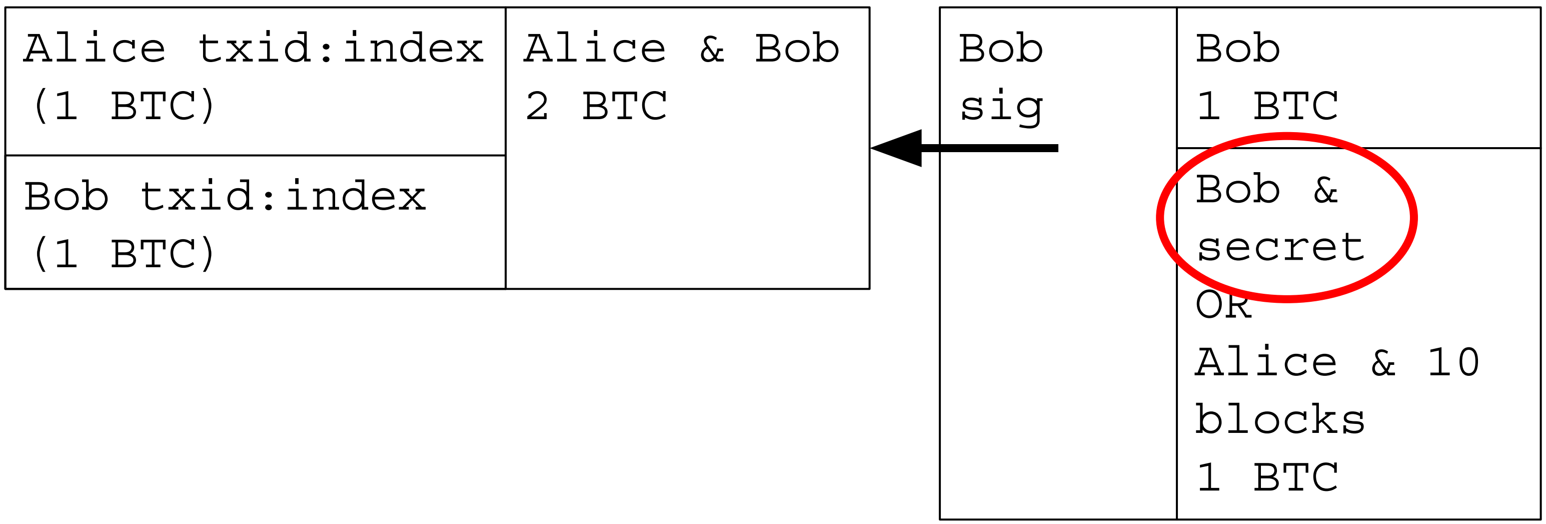
| | |
|---|---|
| Bob<br>sig | Bob<br>1 BTC |
| | Bob &<br>**secret**<br>OR<br>Alice & 10<br>blocks<br>1 BTC |

Bob has a mirror tx

# Updating balances

- Alice and Bob construct new commitment txs to reflect updated balances
- Each gives the other the secret (also called a revocation key) for the previous commitment tx
- If someone broadcasts an old commitment tx, the other party can take the revocation path and taking the entirety of the other output!

# Broadcasting a revoked state



Funding tx

| | |
|---|---|
| Alice txid:index (1 BTC) | Alice & Bob 2 BTC |
| Bob txid:index (1 BTC) | |

Alice’s commitment tx

| | |
|---|---|
| Bob sig | Bob 1 BTC |
| | Bob & secret OR Alice & 10 blocks 1 BTC |

# Closing a channel

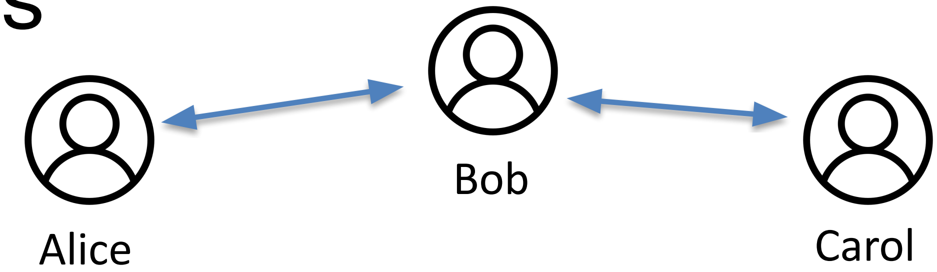
- Cooperative close
  - Both sign a transaction spending to the appropriate balances for each user
- Non-cooperative close
  - Online party broadcasts latest commitment tx
  - Waits for the delay to pass
  - (delay is a chance for other party to contest and take revocation path if this isn’t actually the latest state)

# New role: watchtowers

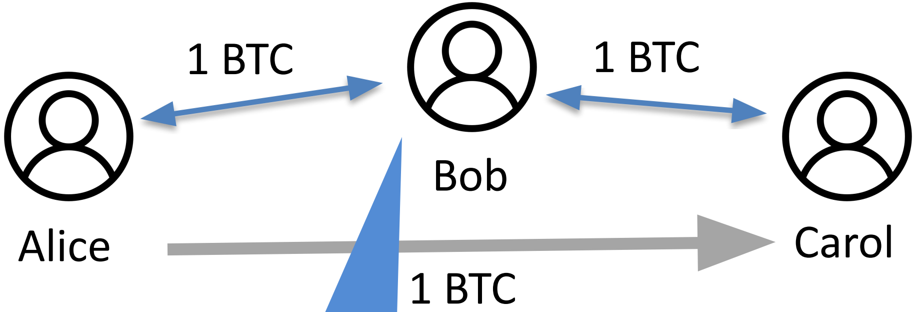
- Users need to be very careful and monitor in case their counterparty broadcasts an old state
- Can outsource this job □ *watchtowers*
- Watchtowers stand at the ready to take action for the user, but can’t spend anything

# Lightning Network

- Too much cost to open a channel with everyone I might want to pay
- Multihop payments: Send a payment *through* other users over multiple channels



# Challenge: make spends on multiple channels atomic

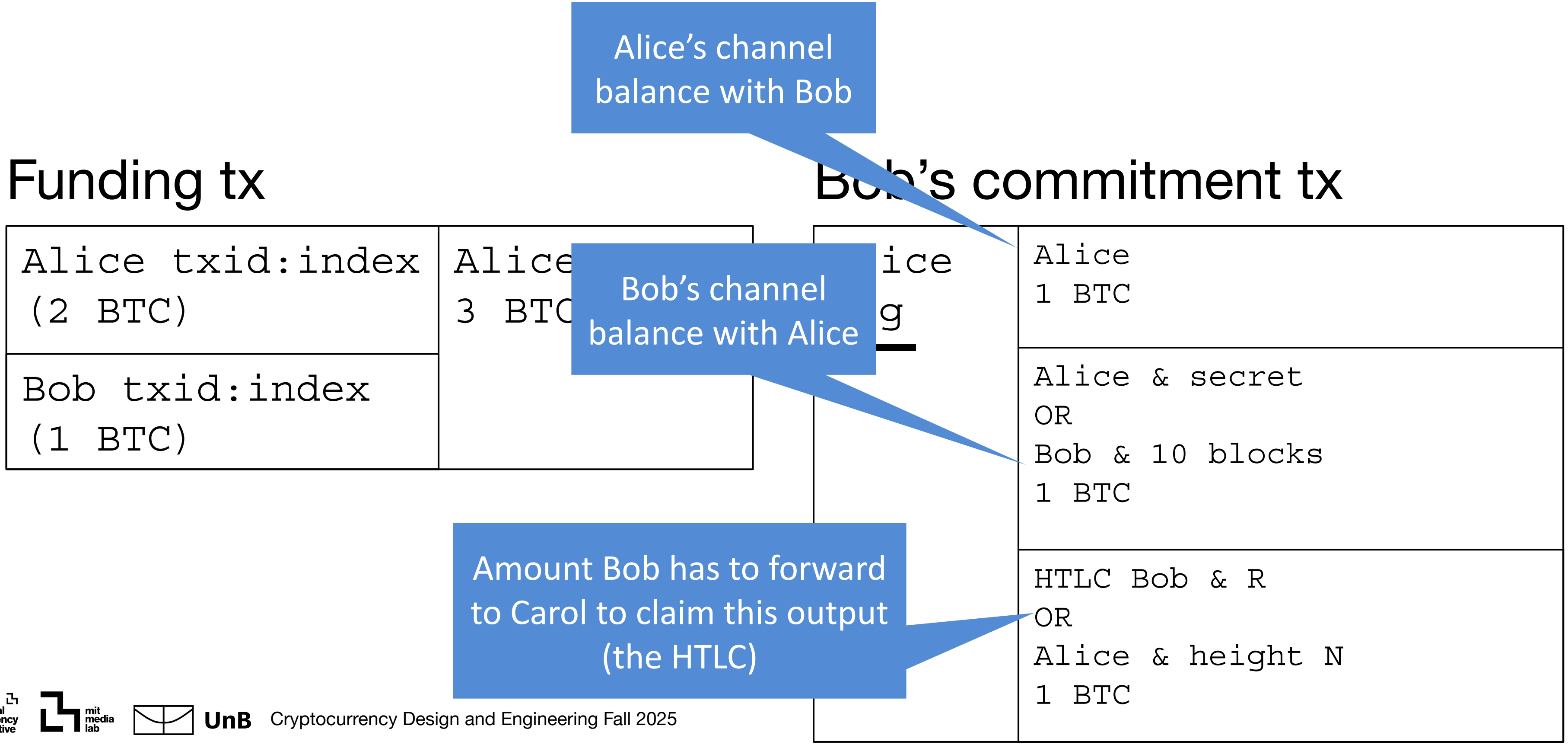


Alice pays 1 BTC to Bob iff Bob pays 1 BTC to Carol

Carol generates secret R, and shares H(R) with Alice

Idea: use HTLCs (Hash Timelock Contracts)

# Alice->Bob HTLC



Funding tx

| | |
|---|---|
| Alice txid:index<br>(2 BTC) | Alice<br>3 BTC |
| Bob txid:index<br>(1 BTC) | |

Bob's commitment tx

| | |
|---|---|
| ice<br>g | Alice<br>1 BTC |
| | Alice & secret<br>OR<br>Bob & 10 blocks<br>1 BTC |
| | HTLC Bob & R<br>OR<br>Alice & height N<br>1 BTC |

Alice's channel balance with Bob

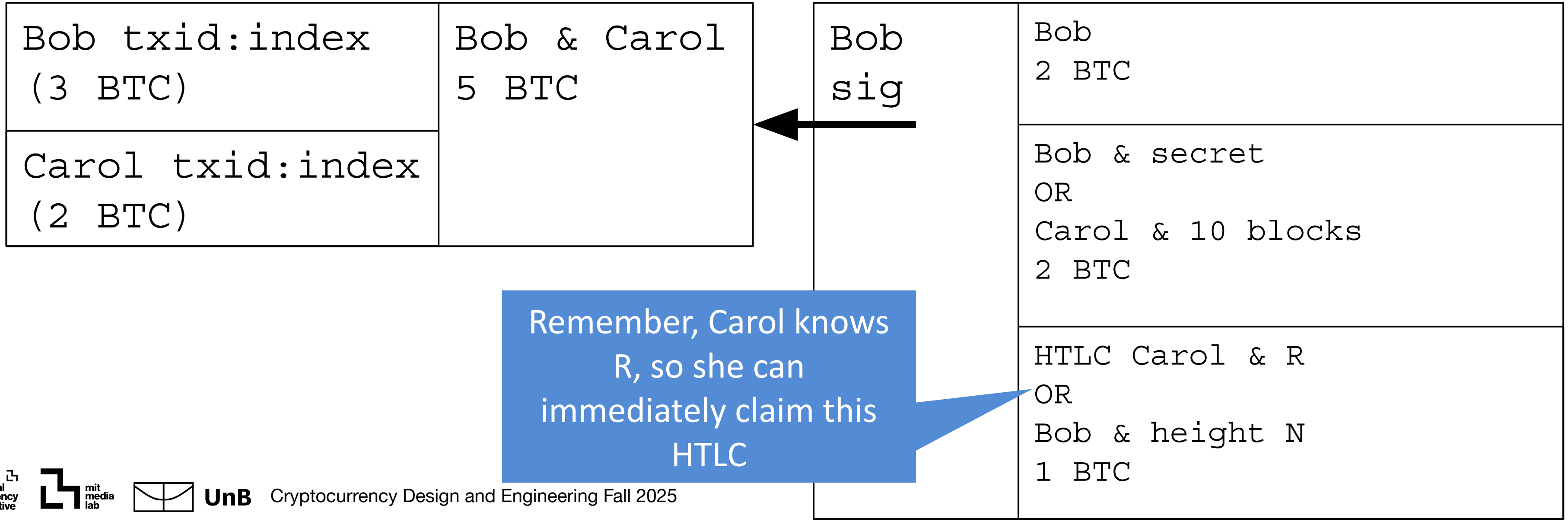
Bob's channel balance with Alice

Amount Bob has to forward to Carol to claim this output (the HTLC)

# Bob->Carol HTLC

Funding tx

Carol’s commitment tx

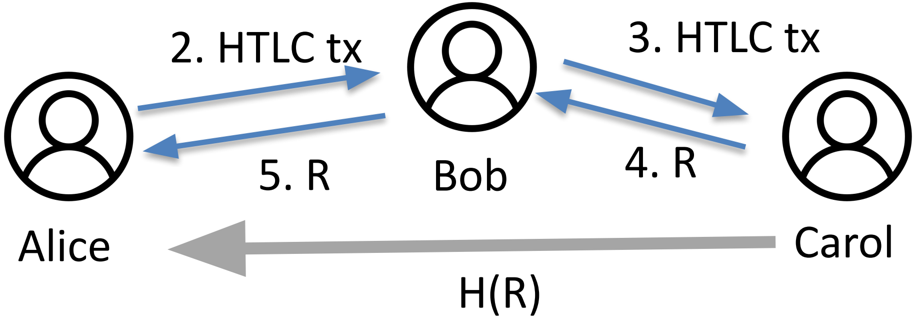


| Inputs | Outputs |
|---|---|
| Bob txid:index (3 BTC) | Bob & Carol<br>5 BTC |
| Carol txid:index (2 BTC) | |

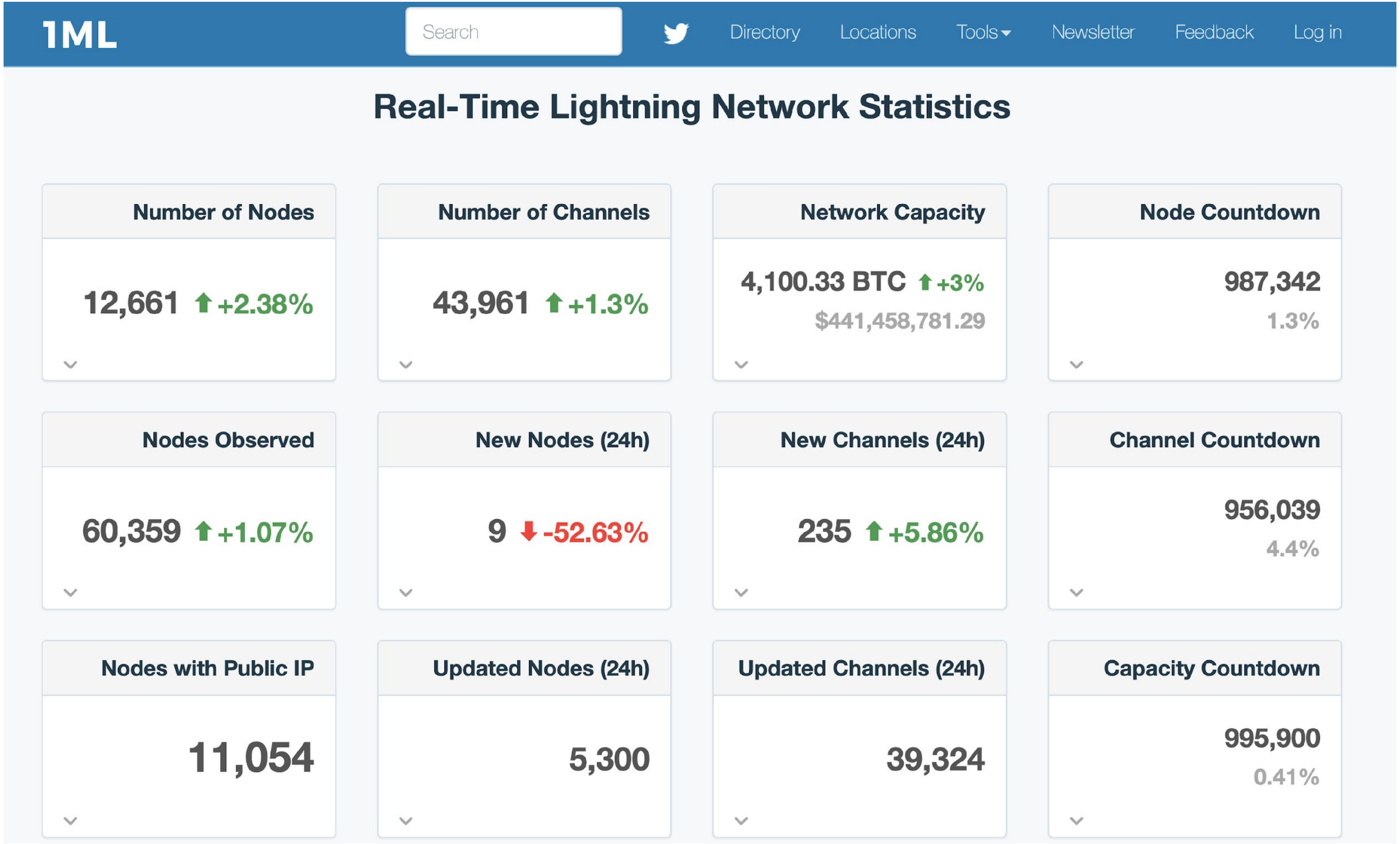
| Inputs | Outputs |
|---|---|
| Bob sig | Bob<br>2 BTC |
| | Bob & secret<br>OR<br>Carol & 10 blocks<br>2 BTC |
| | HTLC Carol & R<br>OR<br>Bob & height N<br>1 BTC |

Remember, Carol knows R, so she can immediately claim this HTLC

# HTLC flow



1. Carol generates secret R, and shares H(R) with Alice



## Real-Time Lightning Network Statistics

| | | |
|---|---|---|
| Number of Nodes | 12,661 | ↑ +2.38% |
| Number of Channels | 43,961 | ↑ +1.3% |
| Network Capacity | 4,100.33 BTC<br>$441,458,781.29 | ↑ +3% |
| Node Countdown | 987,342<br>1.3% | |
| Nodes Observed | 60,359 | ↑ +1.07% |
| New Nodes (24h) | 9 | ↓ -52.63% |
| New Channels (24h) | 235 | ↑ +5.86% |
| Channel Countdown | 956,039<br>4.4% | |
| Nodes with Public IP | 11,054 | |
| Updated Nodes (24h) | 5,300 | |
| Updated Channels (24h) | 39,324 | |
| Capacity Countdown | 995,900<br>0.41% | |

# Lightning pros/cons

**Pros**

- Very good off-chain scalability
- Instant settlement
- Doesn’t require privileged parties

**Cons**

- Additional liveness requirement
- Liquidity management
- Limitations to scaling
- Network complexity
- UX complexity