Emerging Technologies for the Circular Economy

Lecture 9: Consensus Mechanisms and IOTA

Prof. Dr. Benjamin Leiding
M.Sc. Anant Sujatanagarjuna
M.Sc. Shohreh Kia

License

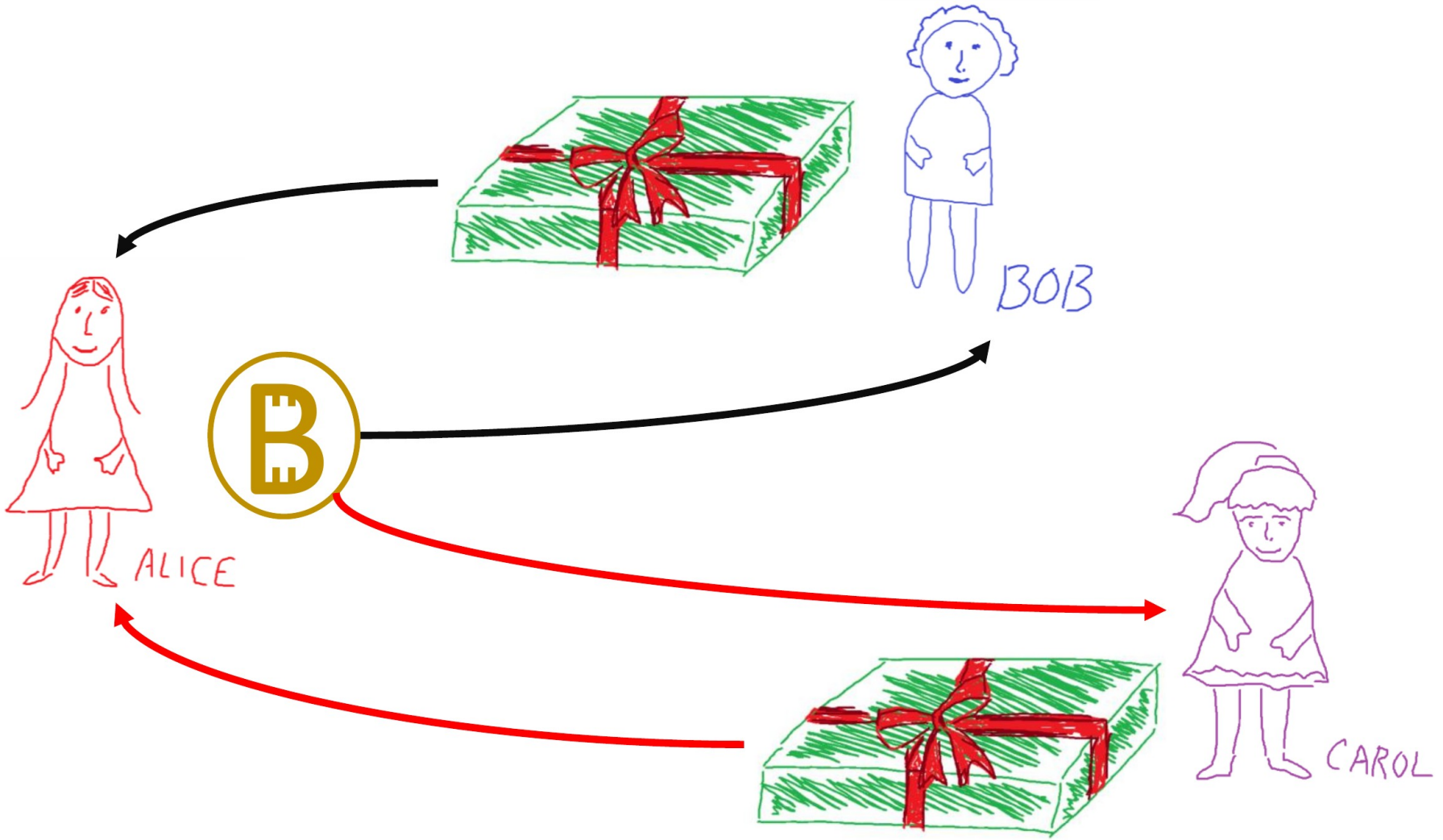
BLOCKCHAIN (RECAP)

Transaction Ledgers

- Essentially a list of transactions
- e.g., Listing money going in/out
- But can also be used for other things, such as maintaining a record of all changes made to a house, all the owners, etc, or an “auto scheckheft”

A german ledger from 1828, by RaphaelQS licensed under CC0 1.0 Universal Public Domain

Double Spending Problem



Double Spending Problem

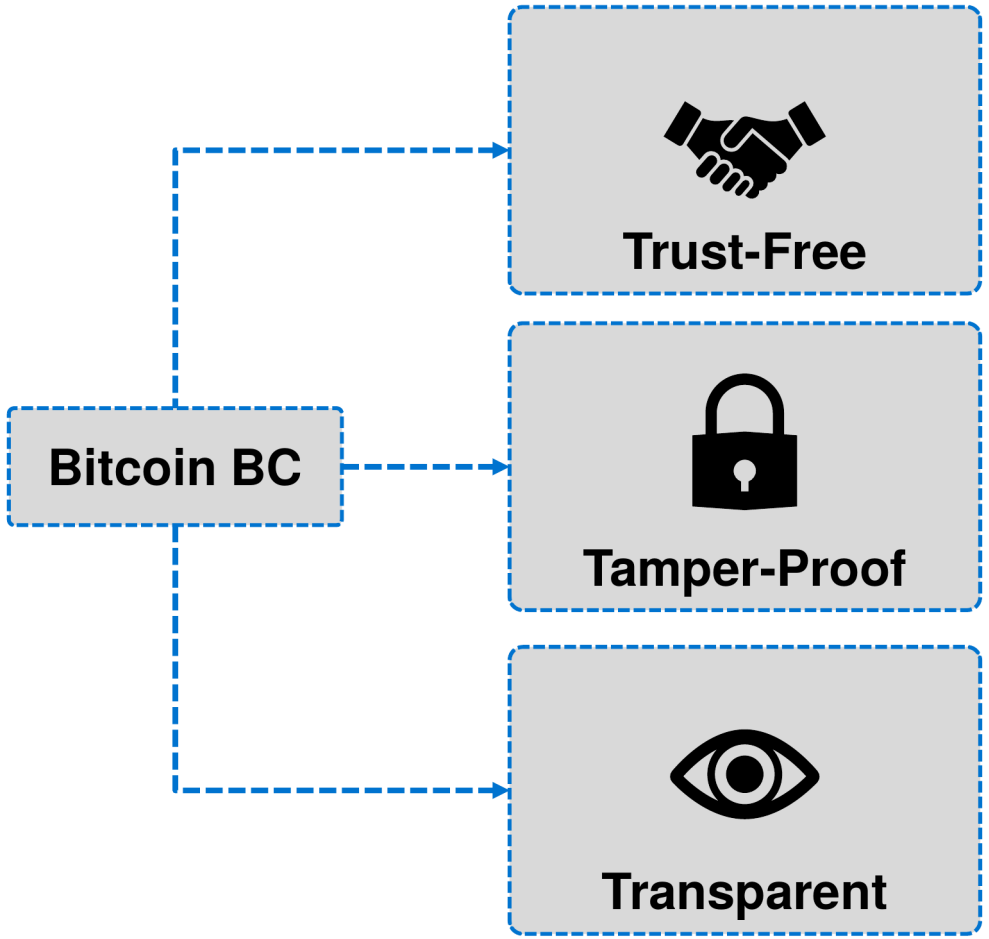
The most common solution to the double spending problem:

- Banks and other intermediaries (Amazon, Ebay) → adds significant cost

- Main problem → trust in intermediaries:
  - The intermediary can tamper with the data
  - Difficult to do on a global scale while maintaining trust, accountability and transparency

Key Properties

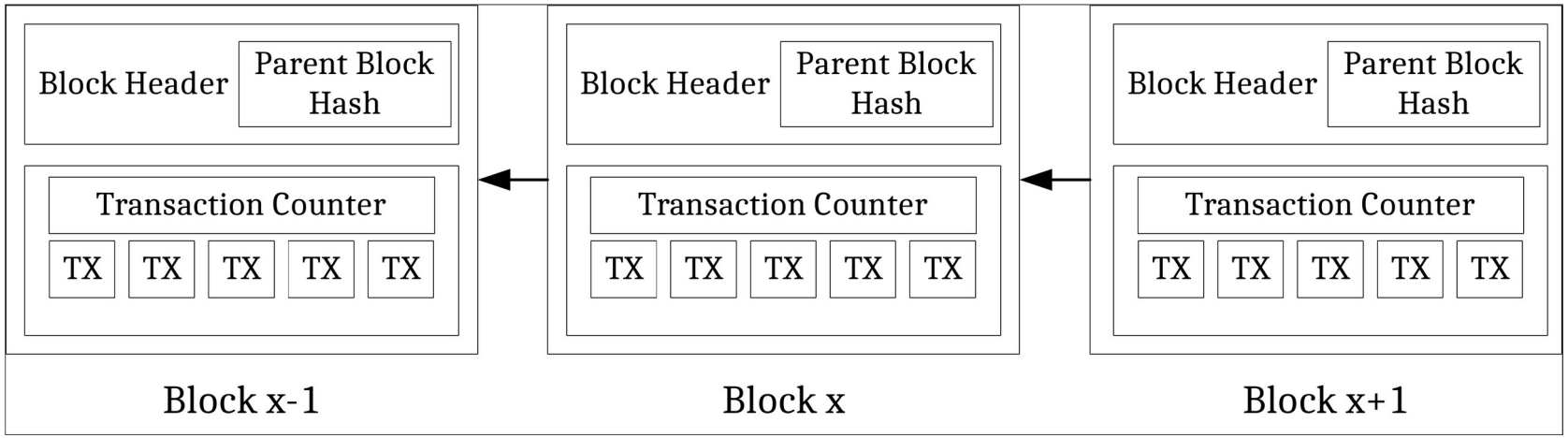
- **Trust-Free:** The system does not require a third party which controls or maintains the system.
- **Tamper-Proof:** The system is resistant to manipulation. The history of events cannot be changed.
- **Transparent:** Every participant of the system can read and validate all information and the current state.



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Blockchain

- Append-only sequential data structure (list)
- New blocks can only be added to the end of the chain
- Chaining blocks together → blockchain
- Changing a block requires recalculation of all subsequent blocks
- Massively duplicated (redundant) across the network (P2P)



BLOCKCHAIN CONSENSUS

Disclaimer and Further Resources

- The lecture slides (Figures are often copied directly) are based on the course Blockchain-based Systems Engineering from TU Munich, which is distributed under a CC-BY-SA 4.0 license
- All their slides, exercises and further information are available online: https://github.com/sebischair/bbse

Distributed consensus

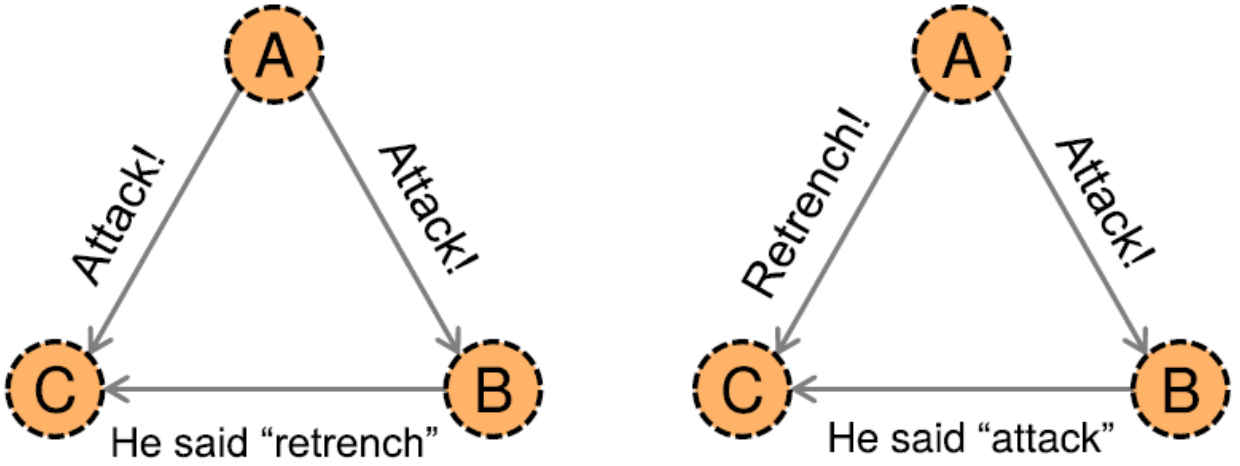
Definition

A network consists out of N nodes. All of these nodes have an input value and propose it to all other nodes. Some of the nodes are faulty (not responding) or malicious, trying to propose a wrong input.

- Two properties must hold:
  - The process has to terminate with all honest nodes in agreement on one input value.
  - The value must have been generated by an honest node.
- Our nodes are trying to agree on the following input:
  - Which of the proposed transactions are valid?
  - In which order do the transactions appear in the ledger?

The Byzantine generals problem

- The Byzantine army wants to invade an enemy city, however, it is separated into multiple divisions. They want to attack at the same time, therefore they have to communicate in between the divisions to and a common time to attack.
- A general is responsible for one division. These generals communicate by messenger. Some of the generals may be traitors, sending wrong messages to other generals. The goal is for all loyal generals to derive the same plan without the traitors being able to convince other generals of the wrong plan.
- It can be shown that if more or equal to one third of the generals are malicious, it is impossible for the honest nodes to derive a common plan. In the figure below, C does not know what to agree on.



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Further difficulties

Problem with the number of nodes:

- Many nodes join the network, also many leave after a short time. How do we know how many there are?
- Approaches like "more than 50% positive votes on a block" would not work, as we do not know how many nodes are in the network.
- How do we prevent an attacker to create an arbitrary number of nodes to increase her chance to be selected? This is called a Sybil-attack.

Problem with time:

- In decentralized networks, there is no general notion of time, as a time server (e.g. NTP) would constitute a central node in the network.
- This makes the development of distributed consensus algorithms difficult.

Bitcoin's approach

Probabilistic consensus
The consensus mechanism is an ongoing process in Bitcoin, therefore the order of blocks or transactions is never 100% safe.

Randomness in consensus
The network selects a random node to propose a new block. As we will see later, this ensures that if over 50% are honest, a probabilistic consensus can be reached.

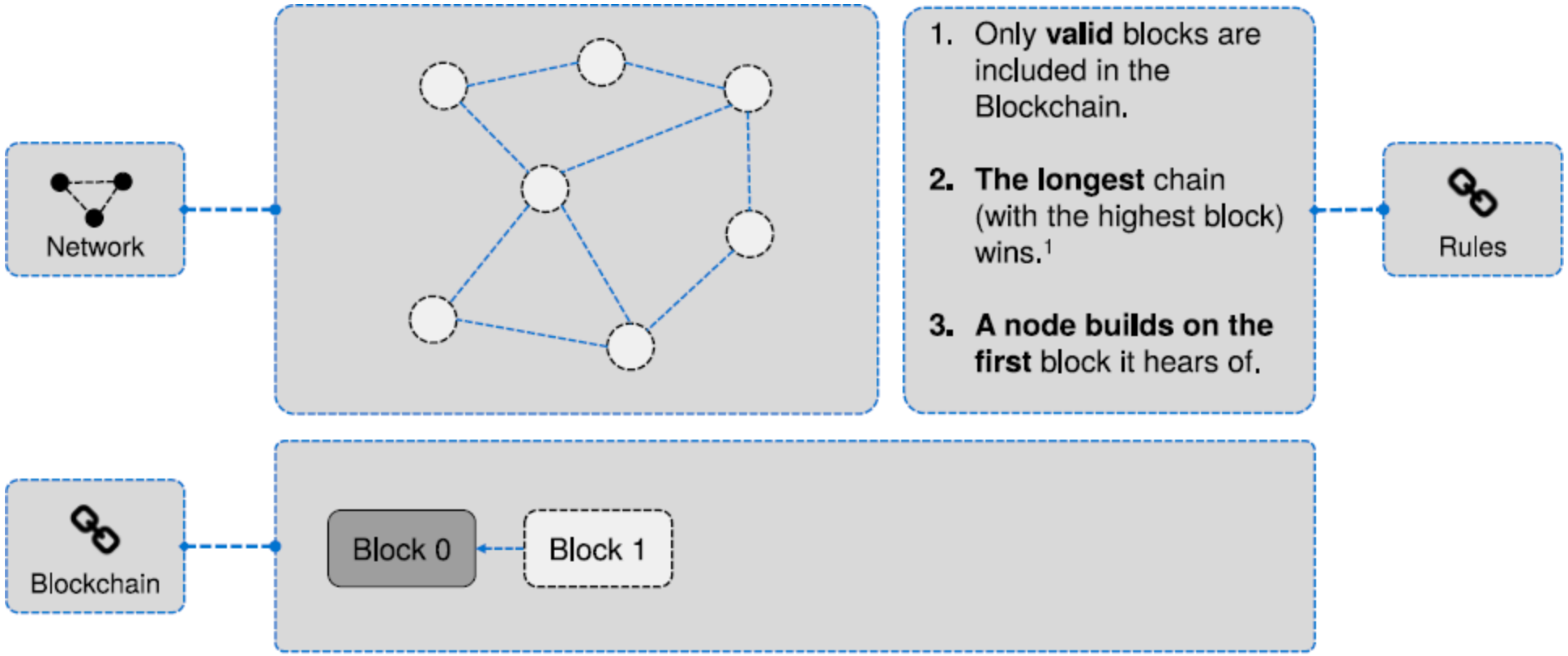
Incentivized nodes
The network incentivizes nodes to participate in the consensus algorithm. They receive bitcoins for created blocks which are included in the longest chain.

U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Simplified consensus of Bitcoin

- **Transaction broadcast:** Every node who receives transactions or creates them, broadcasts them to the network, making everyone aware of new transactions.
- **Block building:** Every node collects the valid transactions, orders them and creates a new block containing the transactions.
- **Random node selection:** A node is randomly chosen out of the network. It is able to propose its block to the network.
- **Block validation:** Other nodes receive the block from the randomly chosen node and validate whether it is correct. A correct block only contains valid transactions.
- **Block acceptance:** Other nodes show their acceptance for this

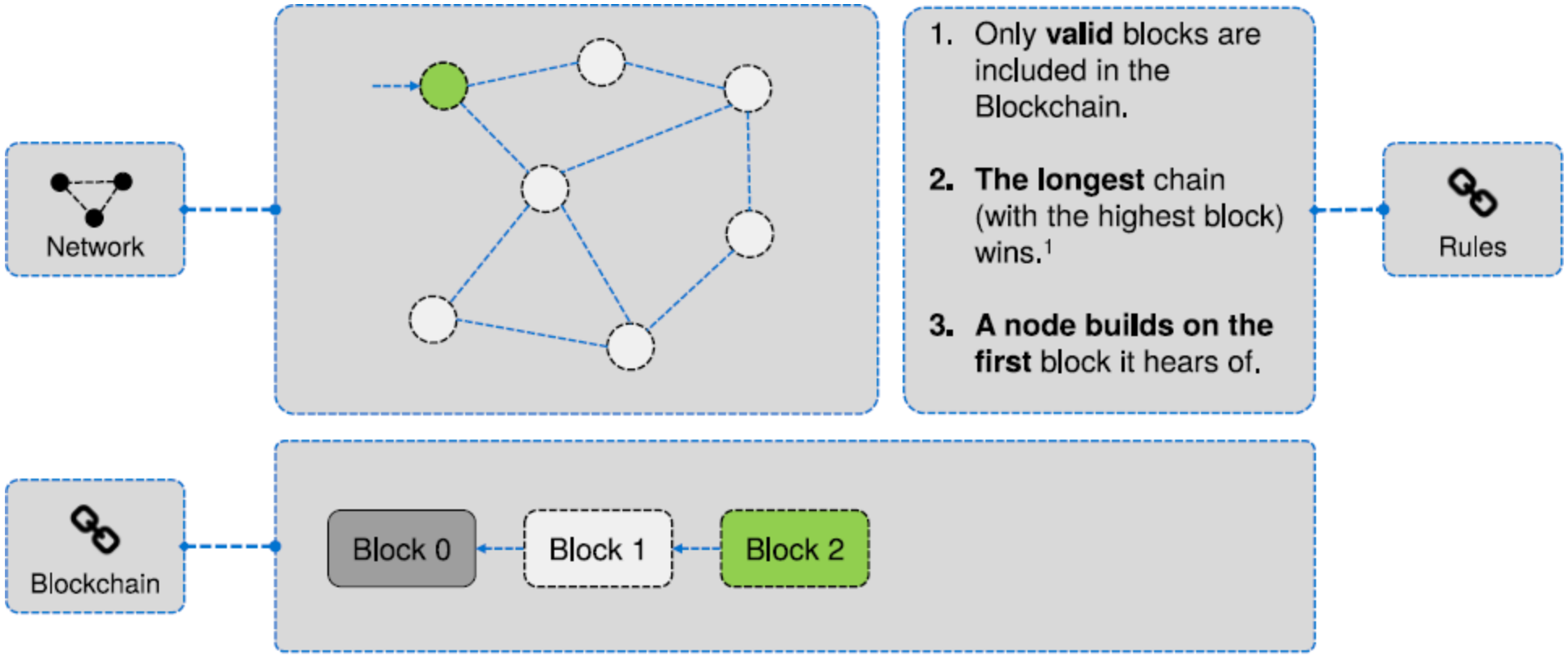
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

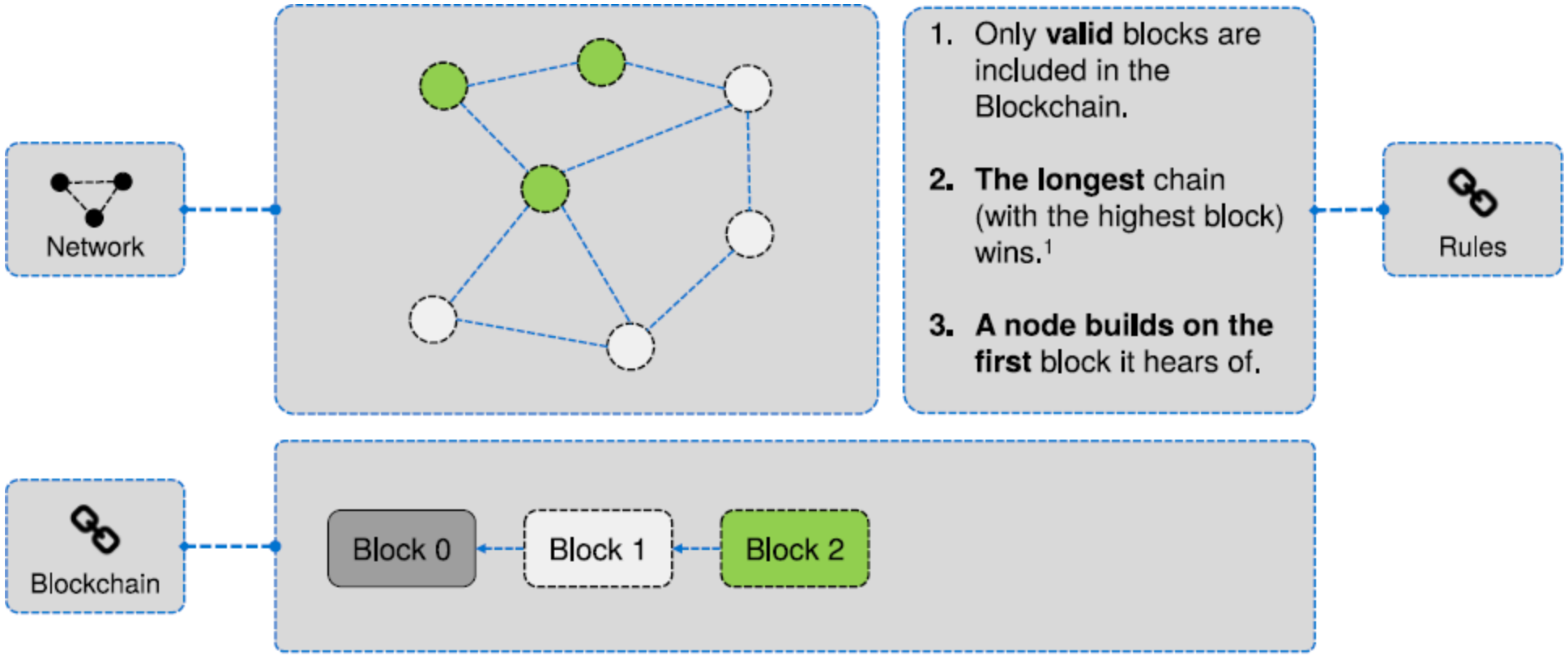
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

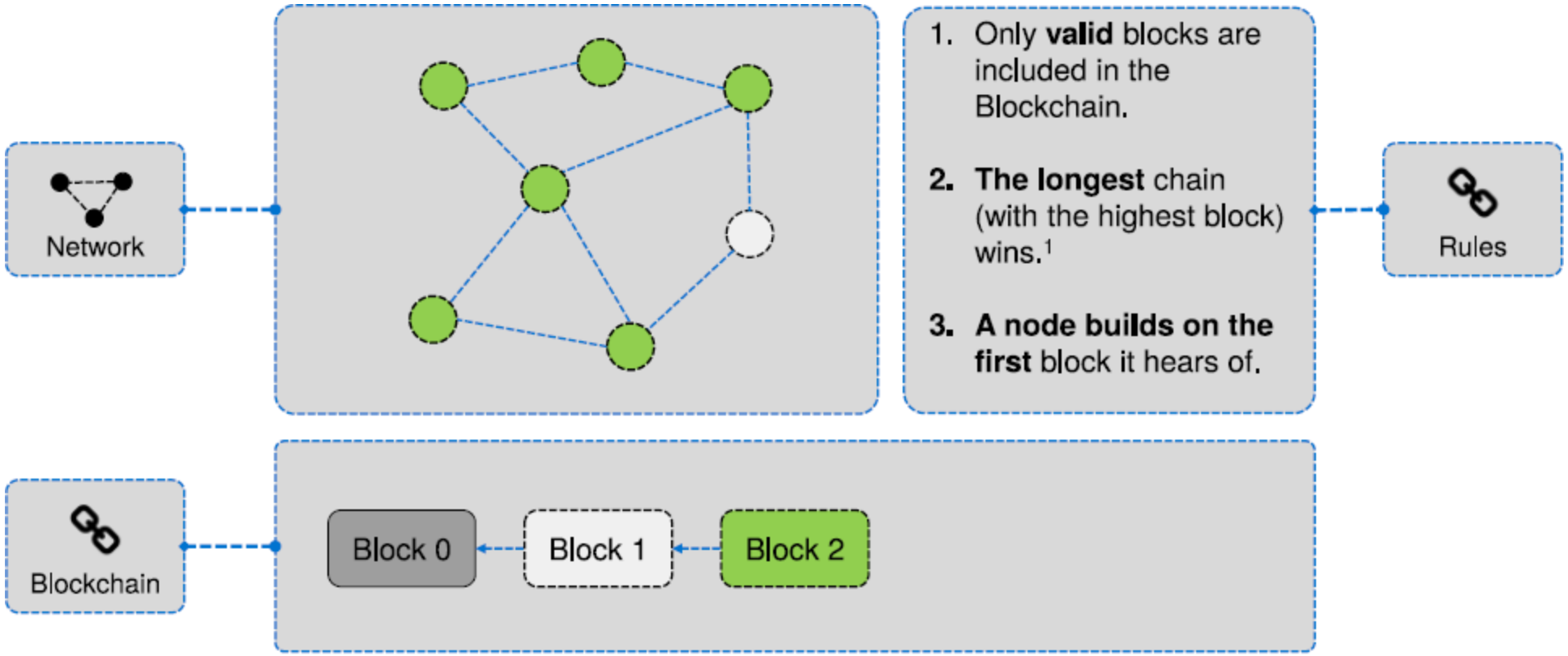
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

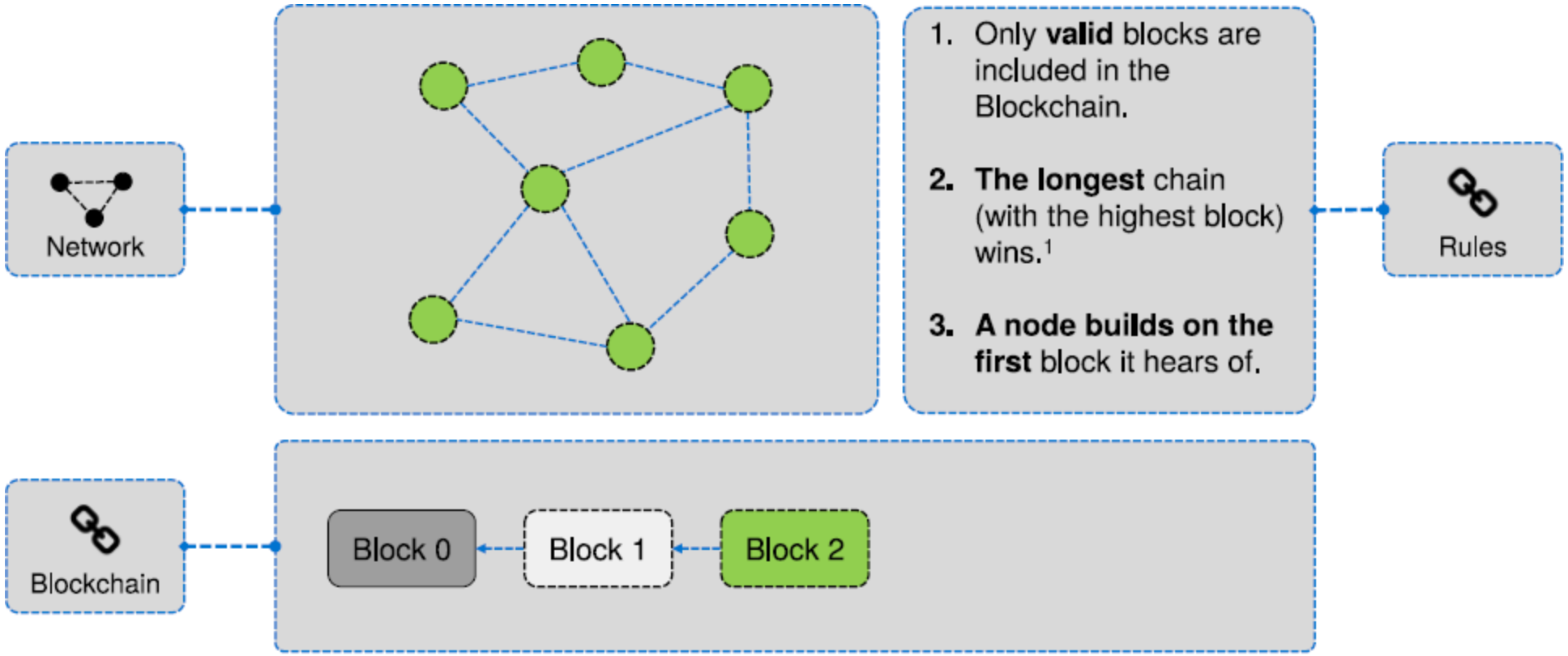
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

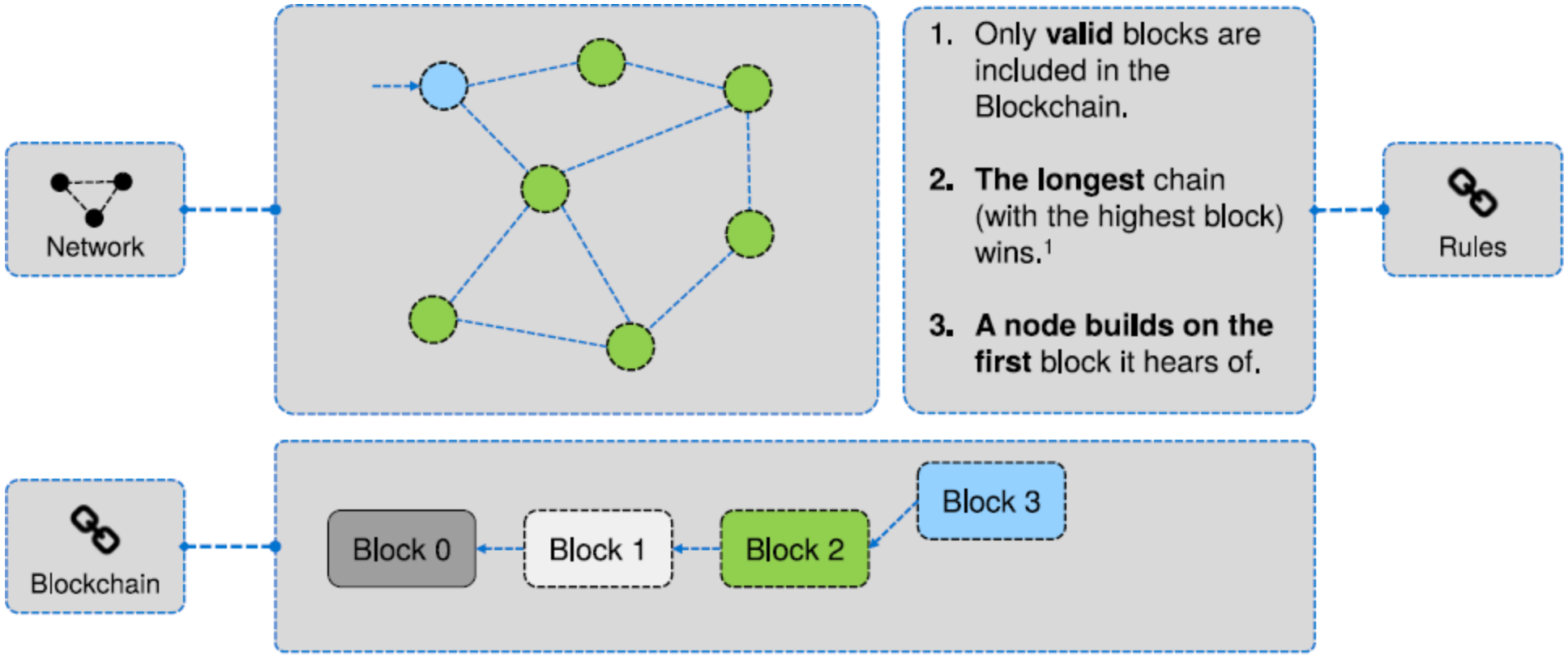
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

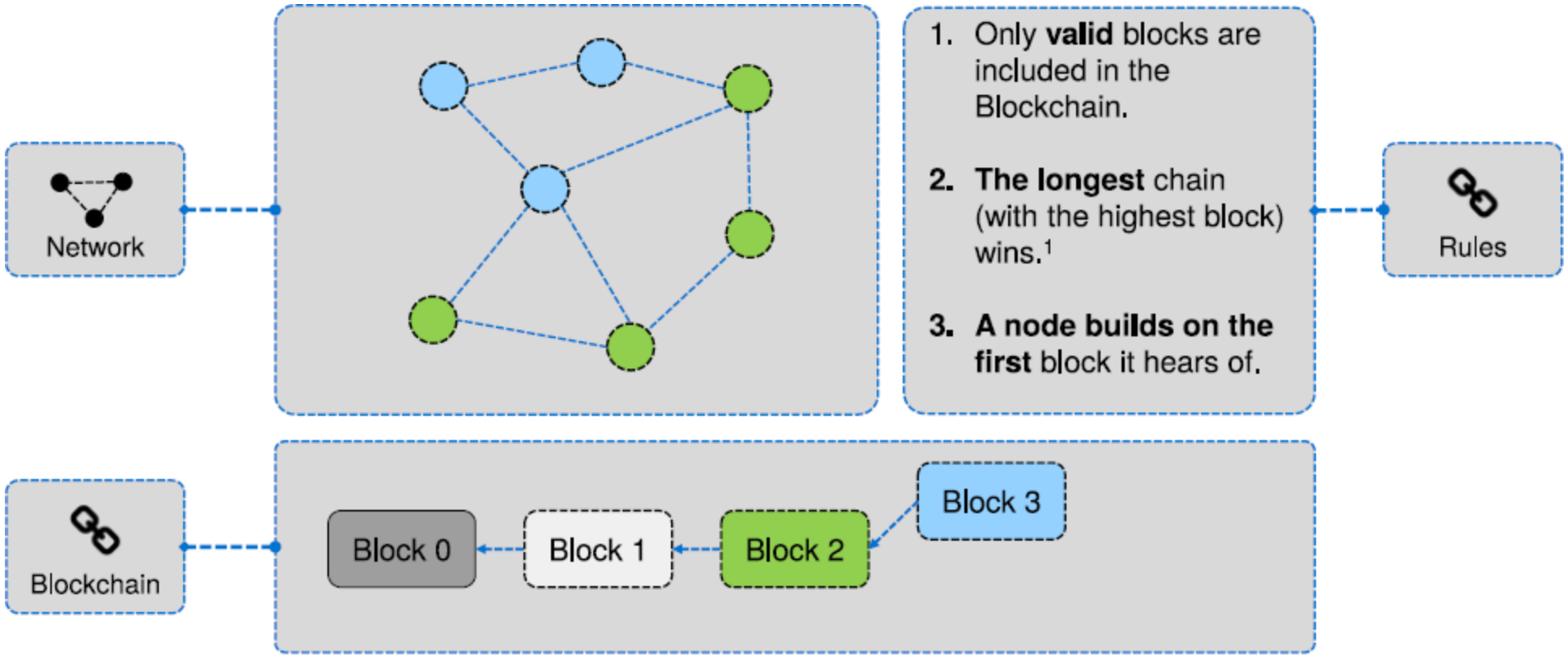
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

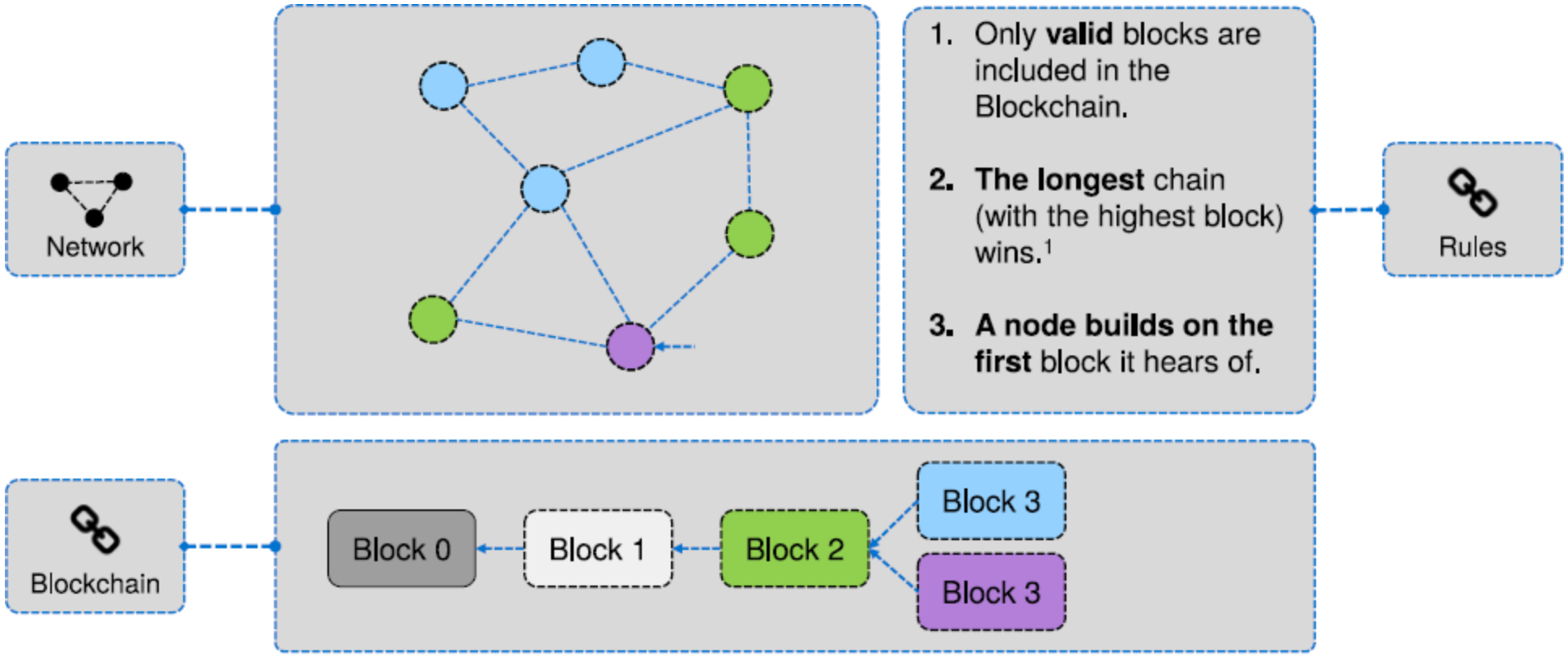
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on,

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

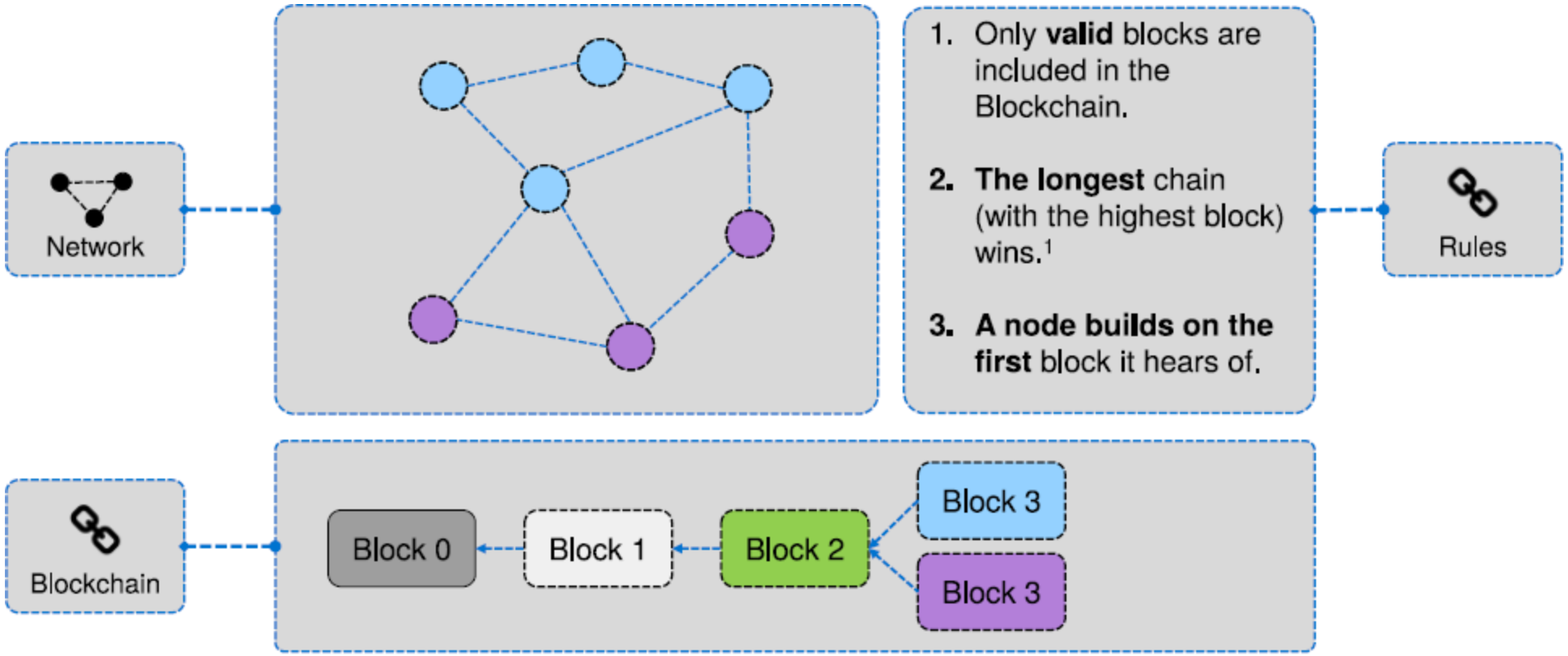
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

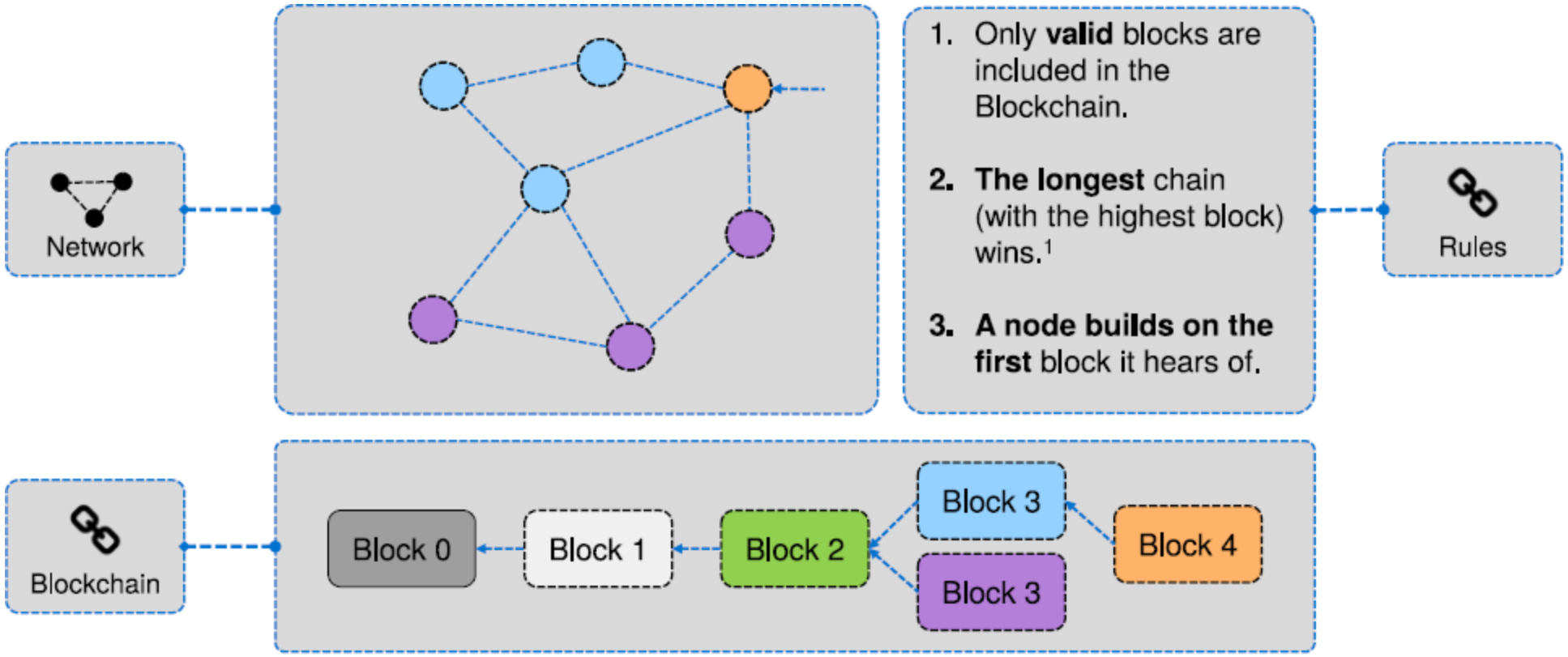
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

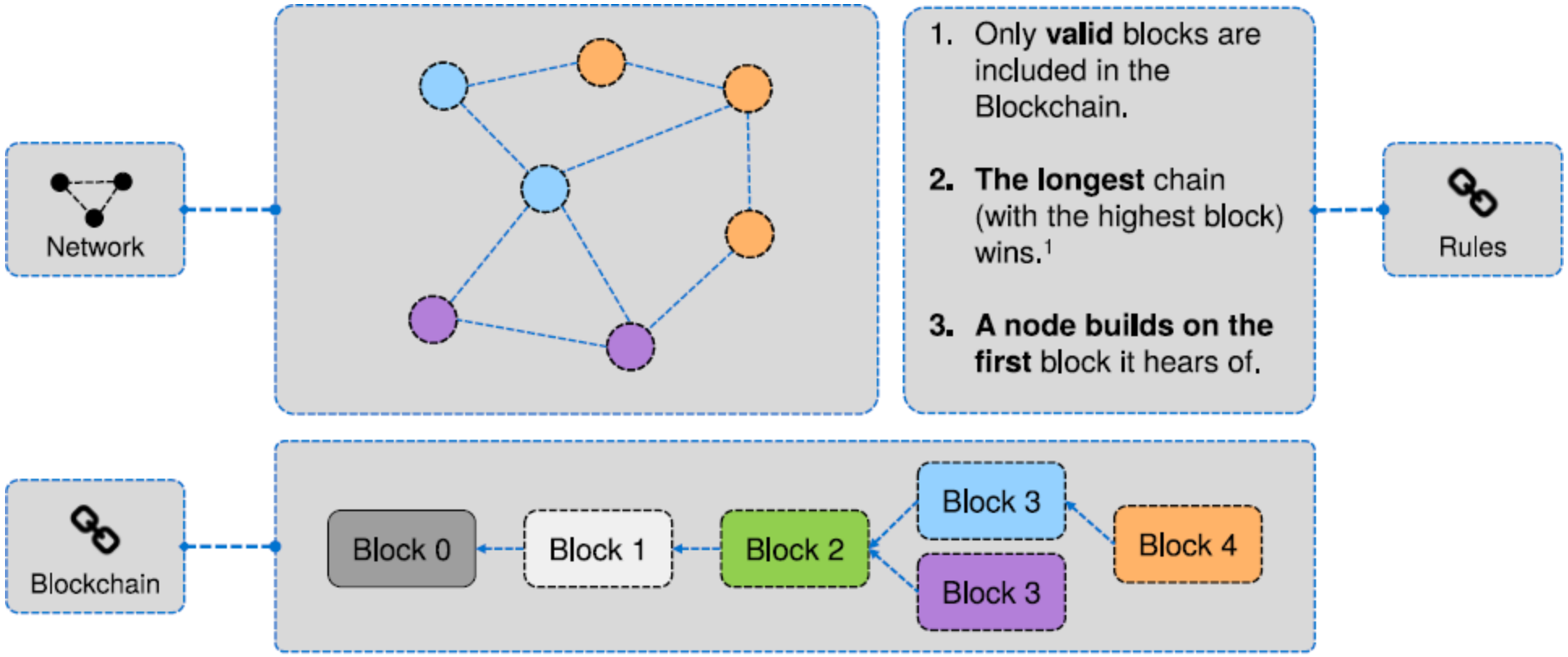
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

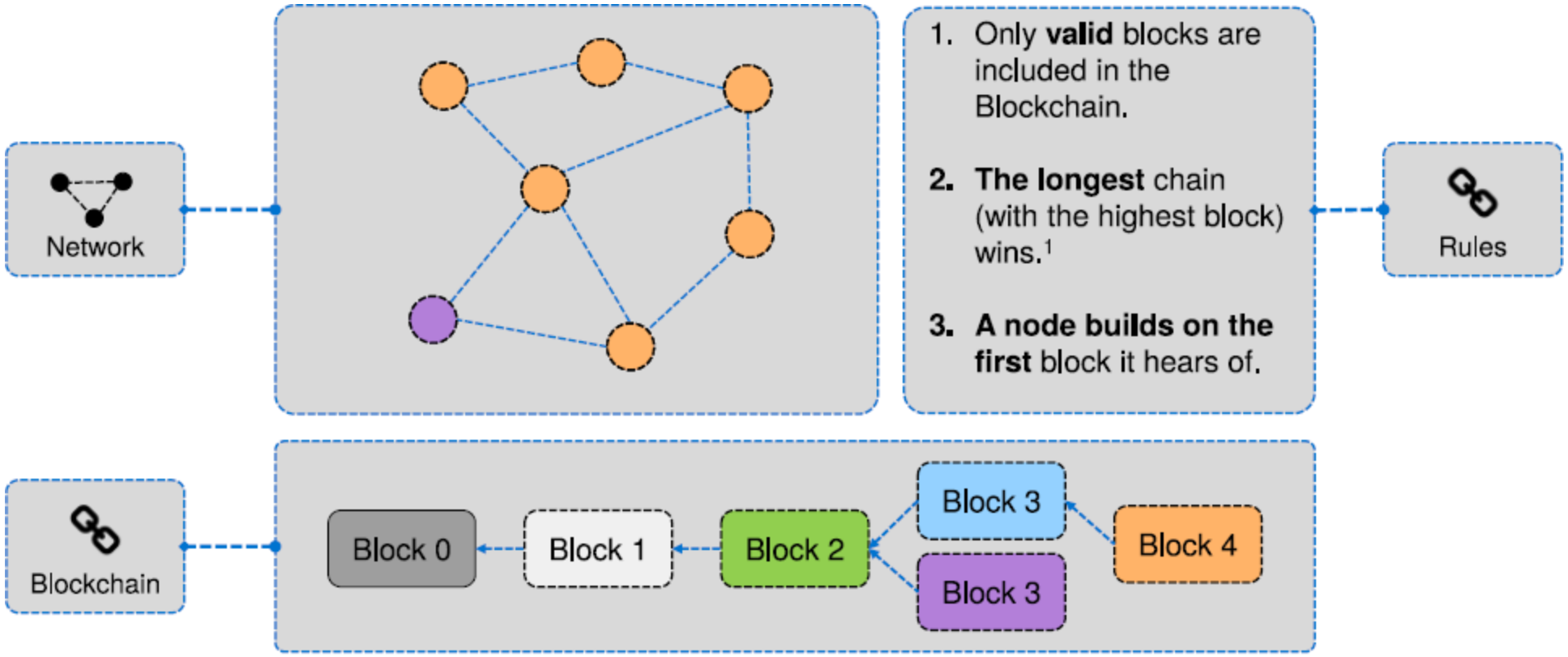
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

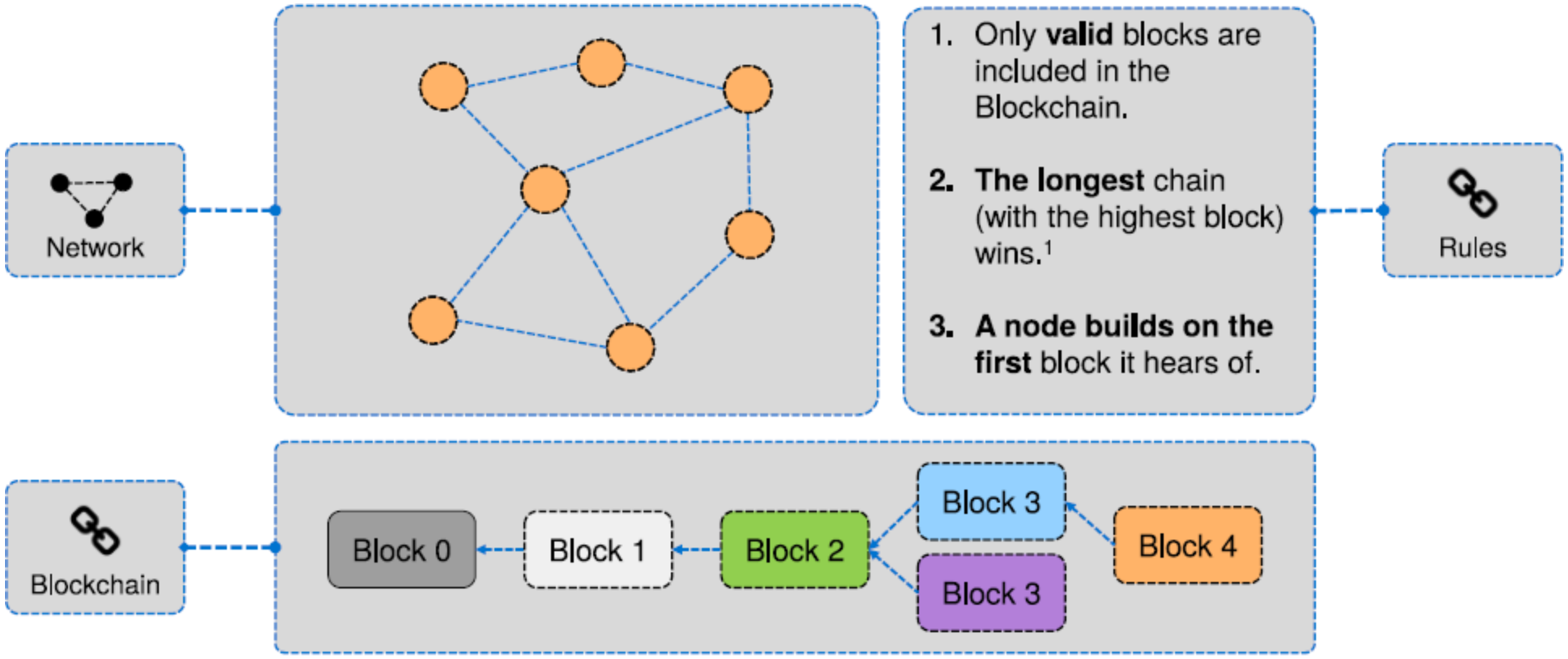
Block propagation



[1] The term "longest chain" is actually the chain with the highest weight, We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Block propagation



[1] The term "longest chain" is actually the chain with the highest weight. We will explain this term later on.

U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

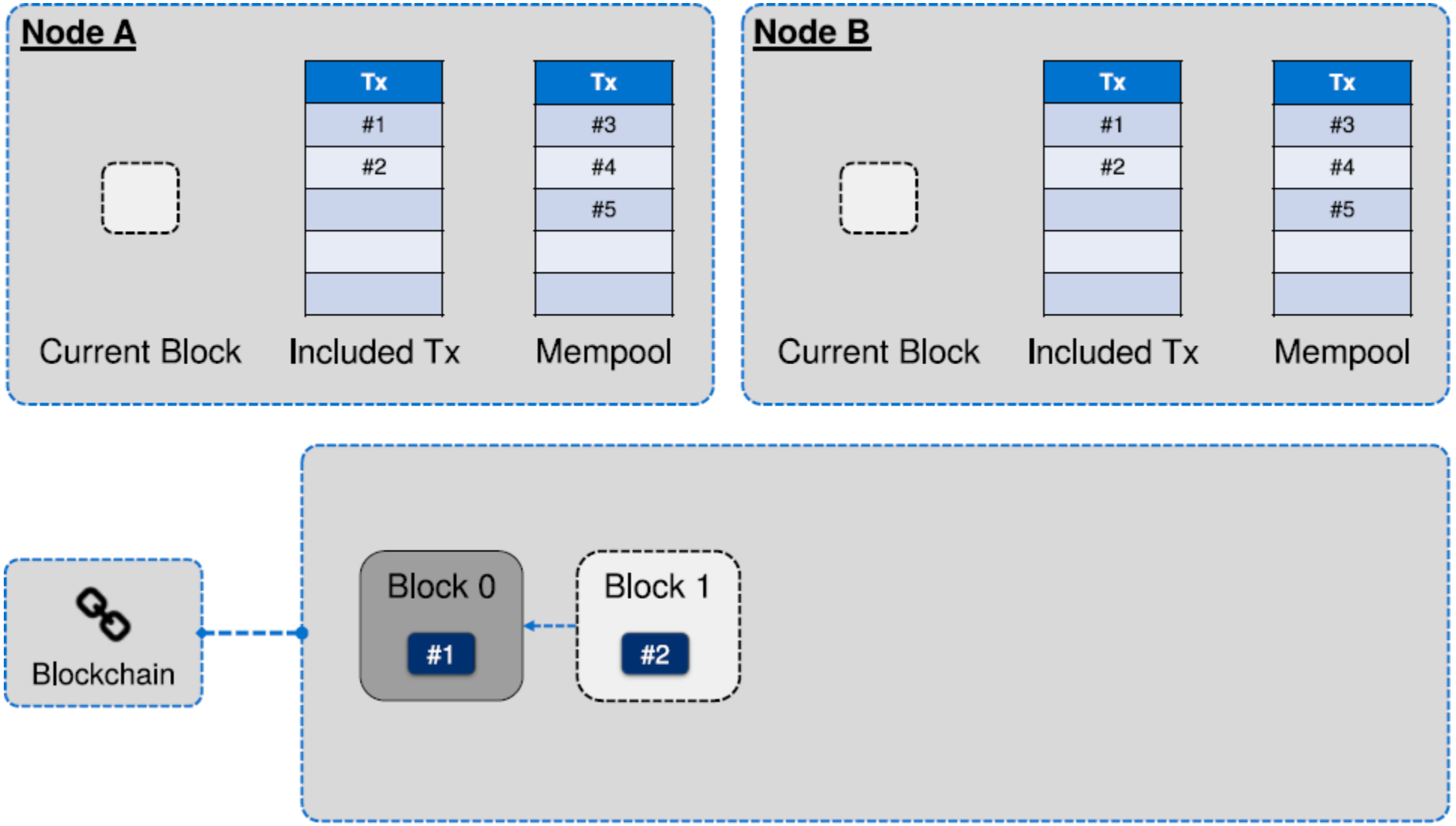
Orphan blocks

Blocks that end up not being used by the longest chain are called orphan blocks.

Now the question arises: What happens to transactions that were mined into orphan blocks?

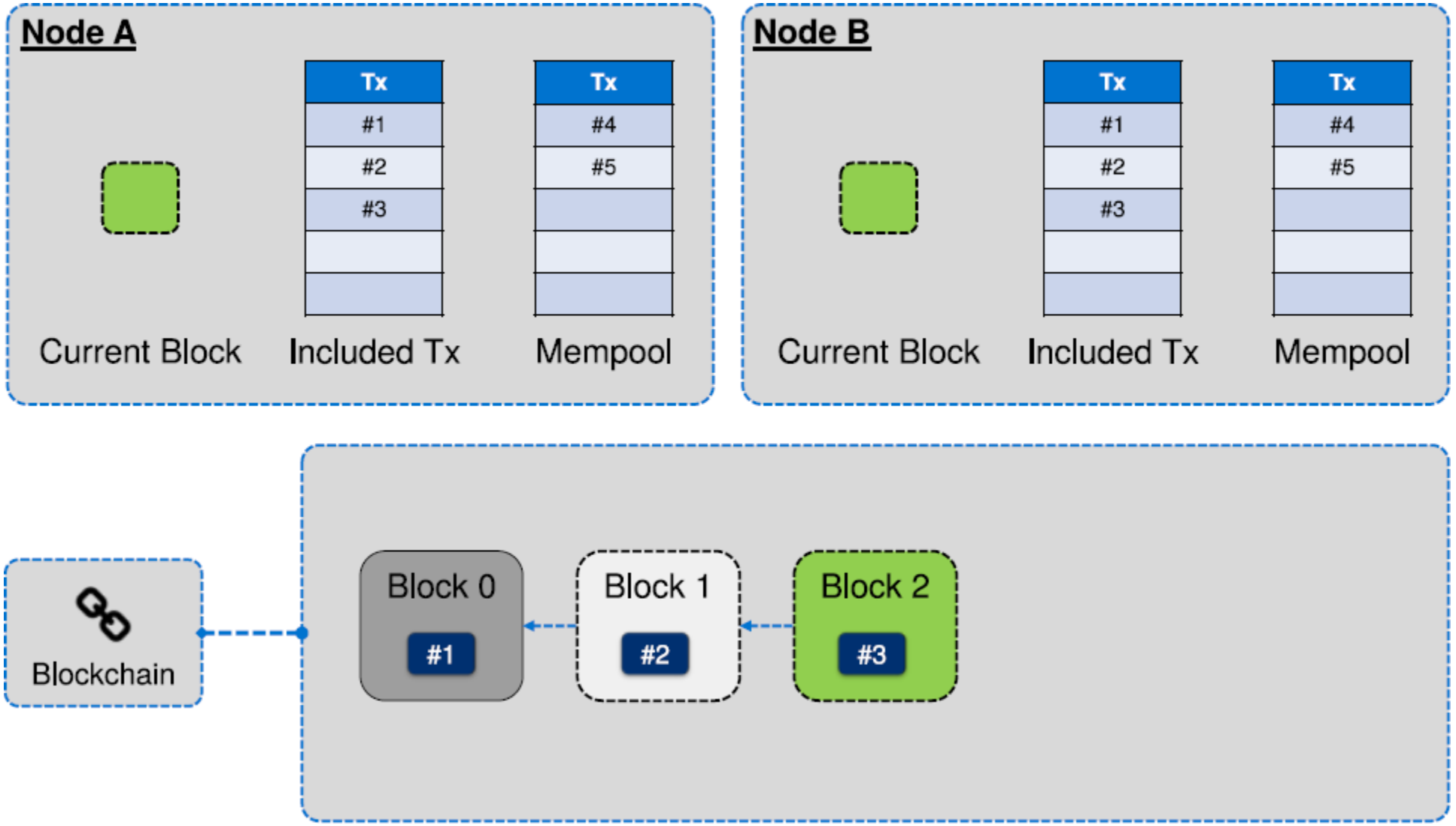
- Unconfirmed transactions are stored in the mempool before they get added to a block.
- As unconfirmed transactions get “gossiped" in the network, every node will know of all transactions.
- As a new block is proposed, all nodes update their mempool and remove the transactions which were included.
- As a consequence, the transactions in an orphan block are simply considered as unconfirmed by those nodes holding a chain not including the block, waiting to be included in a later block.

Orphan block example



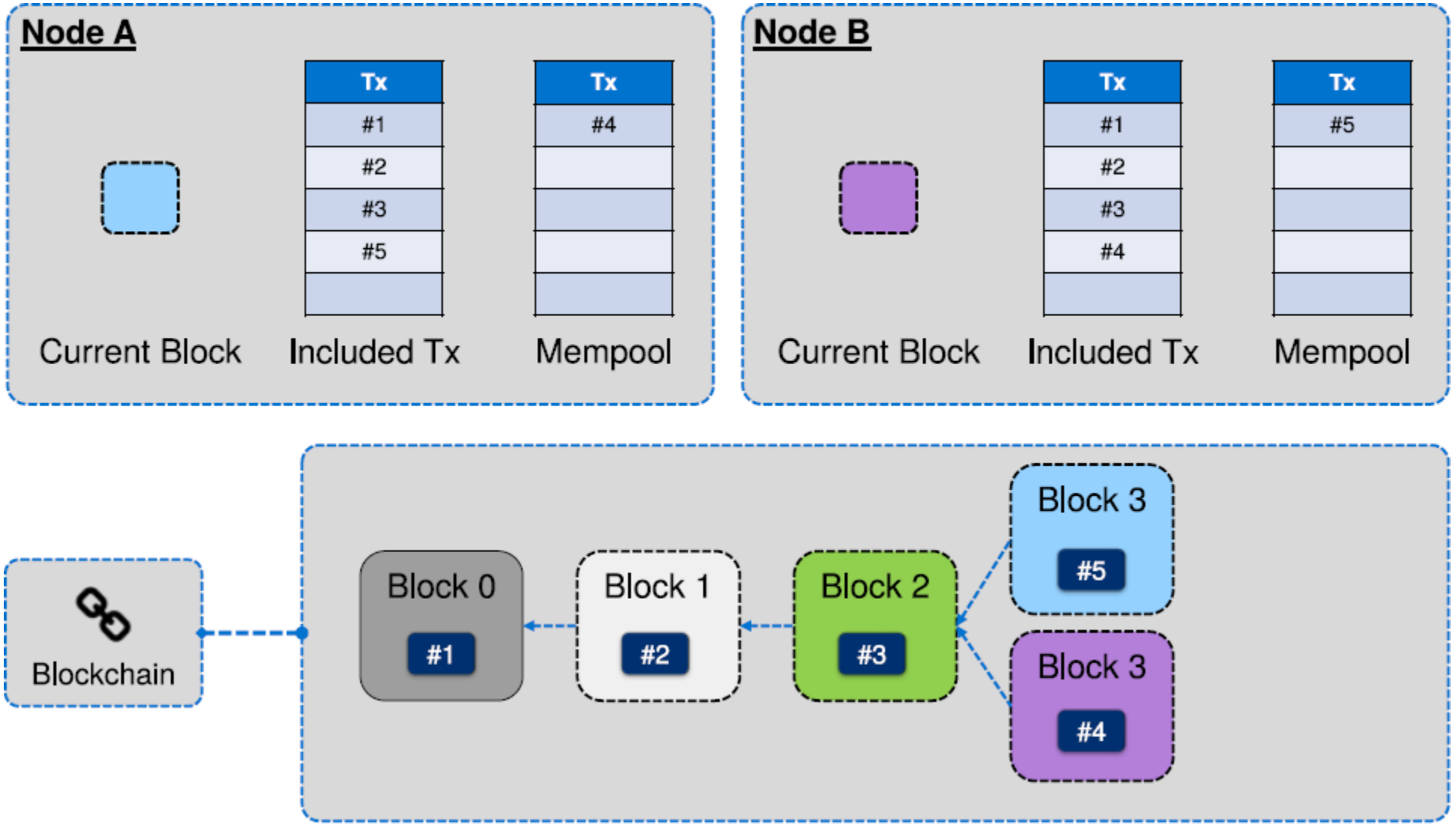
U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Orphan block example



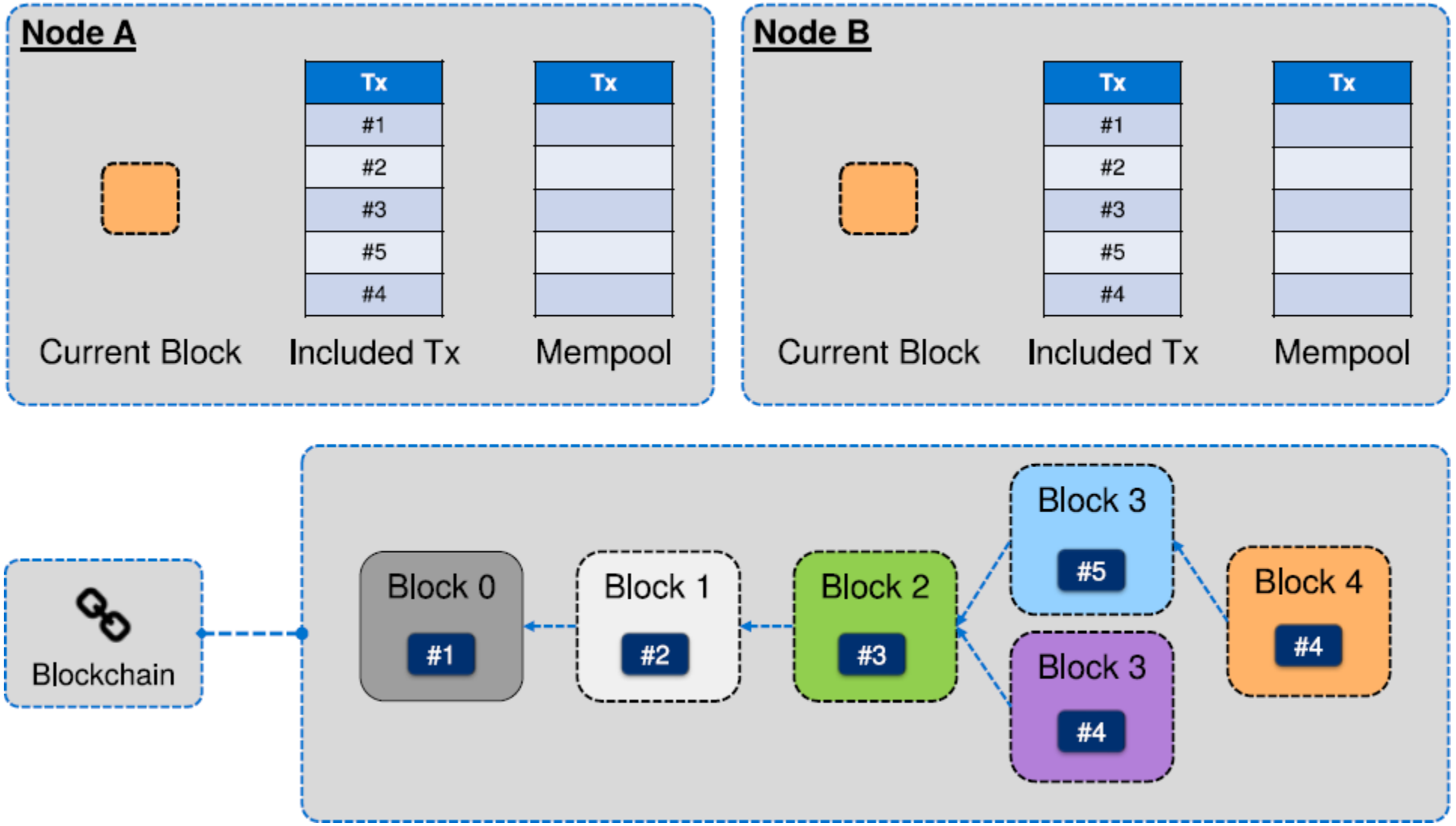
U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Orphan block example



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Orphan block example



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Attacks

- Is it possible to steal bitcoins with an attack against the consensus mechanism?
  - No. Since unspent transaction outputs (UTXO) are secured with the hash of the public key of a user1, an attacker cannot generate a valid transaction spending them.

- Is it possible to censor transactions?
  - Yes. As long as an attacker mines all blocks, they can censor any transaction. If however blocks get mined by non-malicious nodes, these transactions will finally be included in blocks.

Double spending

The idea of digital cash did evolve around the idea that we need to prevent users from spending the same funds multiple times. In the following, we will discuss a scenario where, even with Bitcoin's consensus system, double spends remain possible.

Consider the situation that Alice sends funds to Bob to pay for a digital version of a song. Upon receiving the transaction, Bob sends the song to Alice.

However, at the same time, Alice disseminates a transaction spending the same funds to herself starting from a different node in the network, but Bob does not see this transaction, because he received the other transaction first and the nodes near him to not forward the invalid, second transaction to him.

At this point, multiple things may occur:

- The transaction spending the money to Bob gets mined first and Bob actually receives his money.
- The double spend transaction gets mined first and Alice gets her money back, having gotten the song for free.
- Both transactions get mined into two different blocks. Depending on which of the blocks ends up in a longer chain, either of the previous cases may happen. The other side will end up being orphaned and no longer relevant.

Double spending

Does this mean that double spending is possible?

A block is not valid if it contains conflicting (double spending) transactions, but in certain cases the view on the current consensus may differ between nodes for a short period of time. To mitigate this, it is common for vendors to wait for a certain number of "confirmations" or number of blocks at the top of the chain containing the relevant transaction, before considering it to have gone through. For Bitcoin, it is common to wait for six confirmations.

Block creation

This all leads us to the following requirements for our consensus scheme:

- The network agrees upon who creates the next block in a decentralized manner.
- The scheme can tolerate users joining and leaving at any time.
- The scheme must be resistant against Sybil attacks (one entity controlling arbitrary numbers of nodes).

→ Make use of scarce resources.

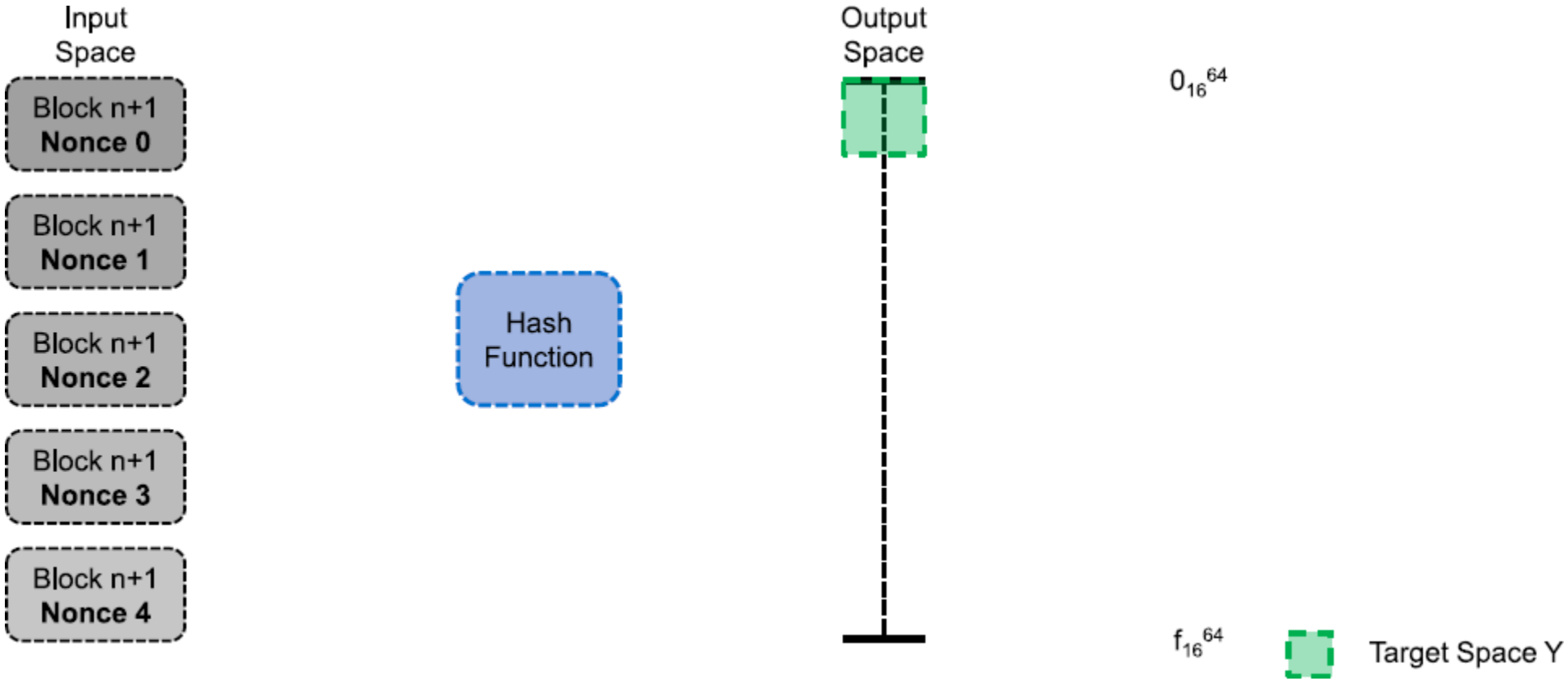
Scarce resources

Proof of Work (PoW): ▪ Based on search puzzles that require large amounts of tries ▪ High investment costs ▪ High energy costs ▪ Leads to arms race ▪ High ongoing attack costs ▪ Anonymous mining ▪ Used by Bitcoin	Proof of Stake (PoS): ▪ Lock up ("stake") coins to get a chance to create the next block ▪ Requires large amount of staked funds ▪ Low energy costs ▪ "The rich get richer" ▪ No ongoing attack costs ▪ Penalties to staked funds my discourage attacks ▪ Not well suited to bootstrap blockchains, due to no initial price **There are also many, many other approaches.**

PROOF OF WORK

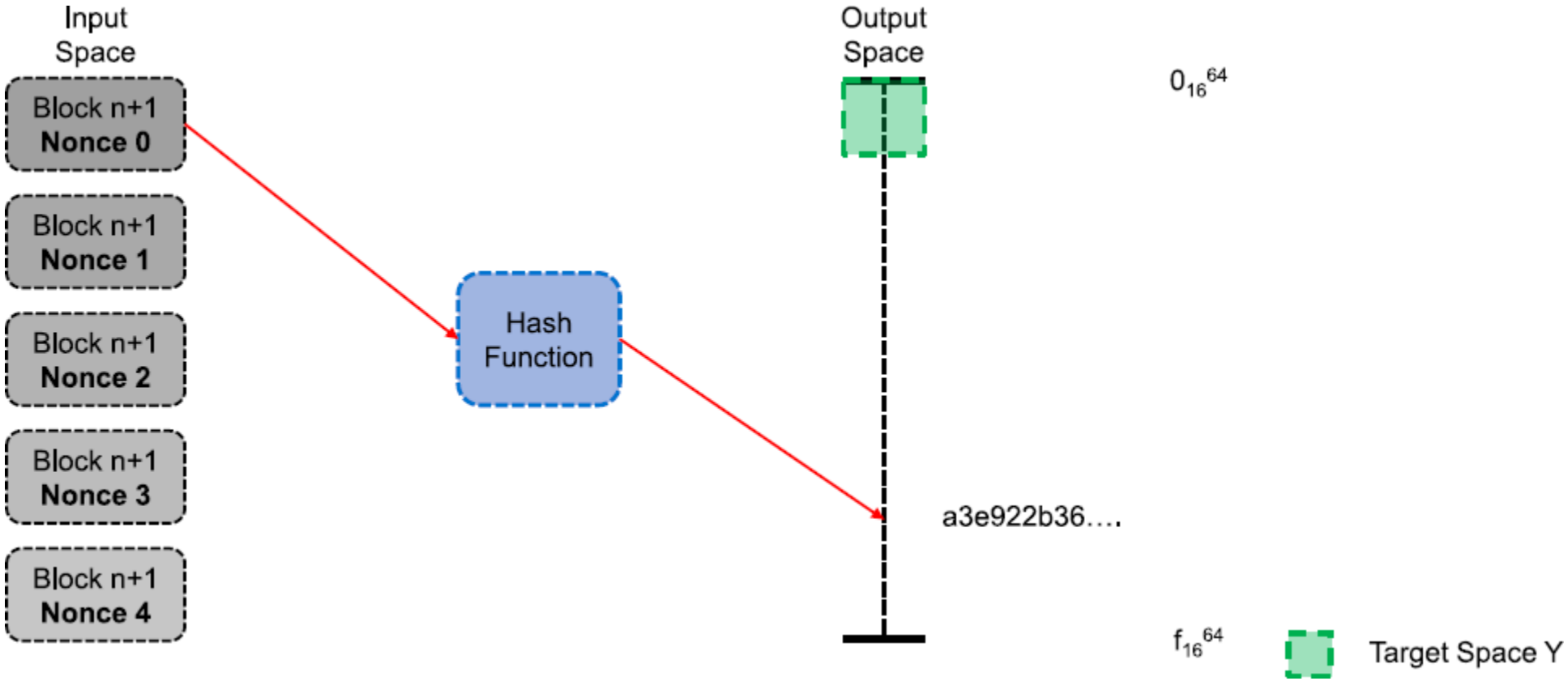
Search puzzle

Idea: We use the search puzzle introduced in the chapter about cryptographic foundations. The header of the hash has to be included in Y. Bitcoin uses double SHA-256. (sha256(sha256(block)))



Search puzzle

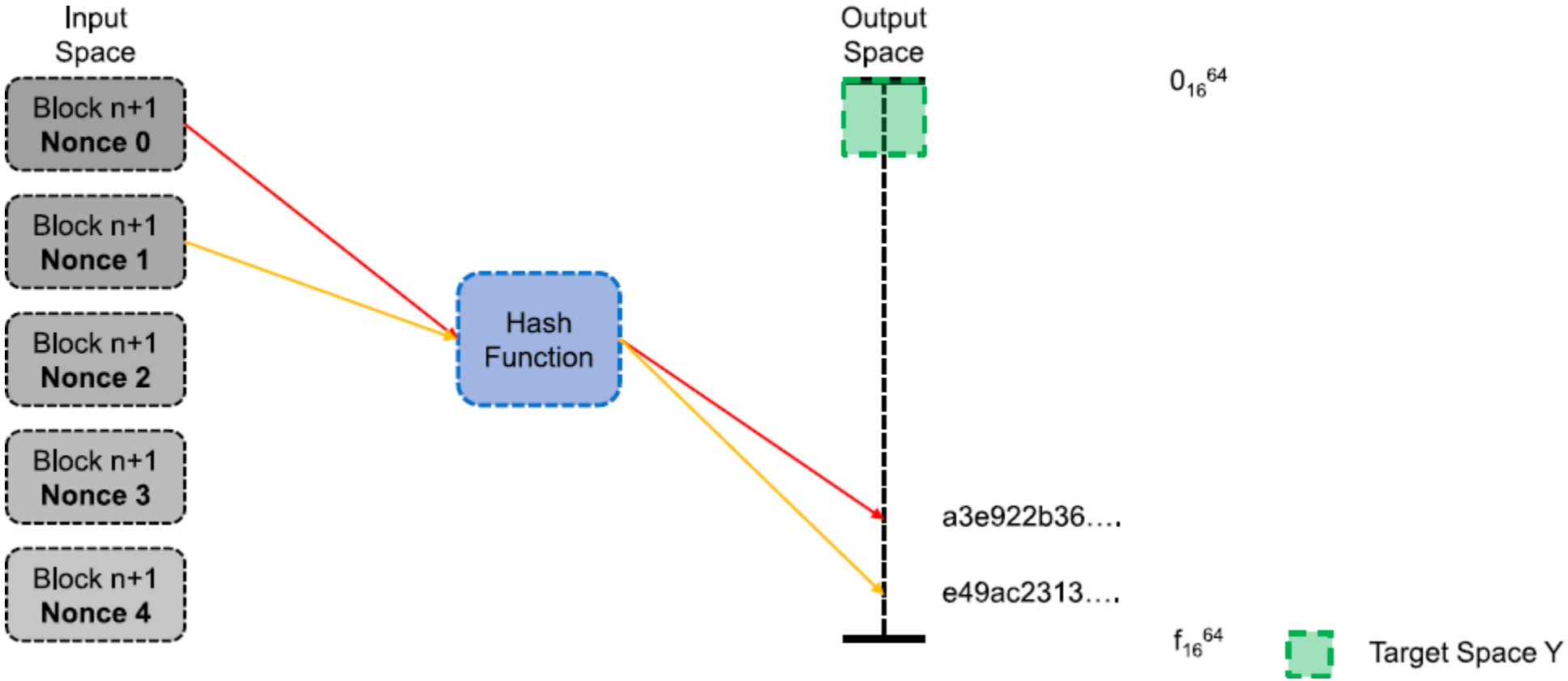
Idea: We use the search puzzle introduced in the chapter about cryptographic foundations. The header of the hash has to be included in Y. Bitcoin uses double SHA-256. (sha256(sha256(block)))



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Search puzzle

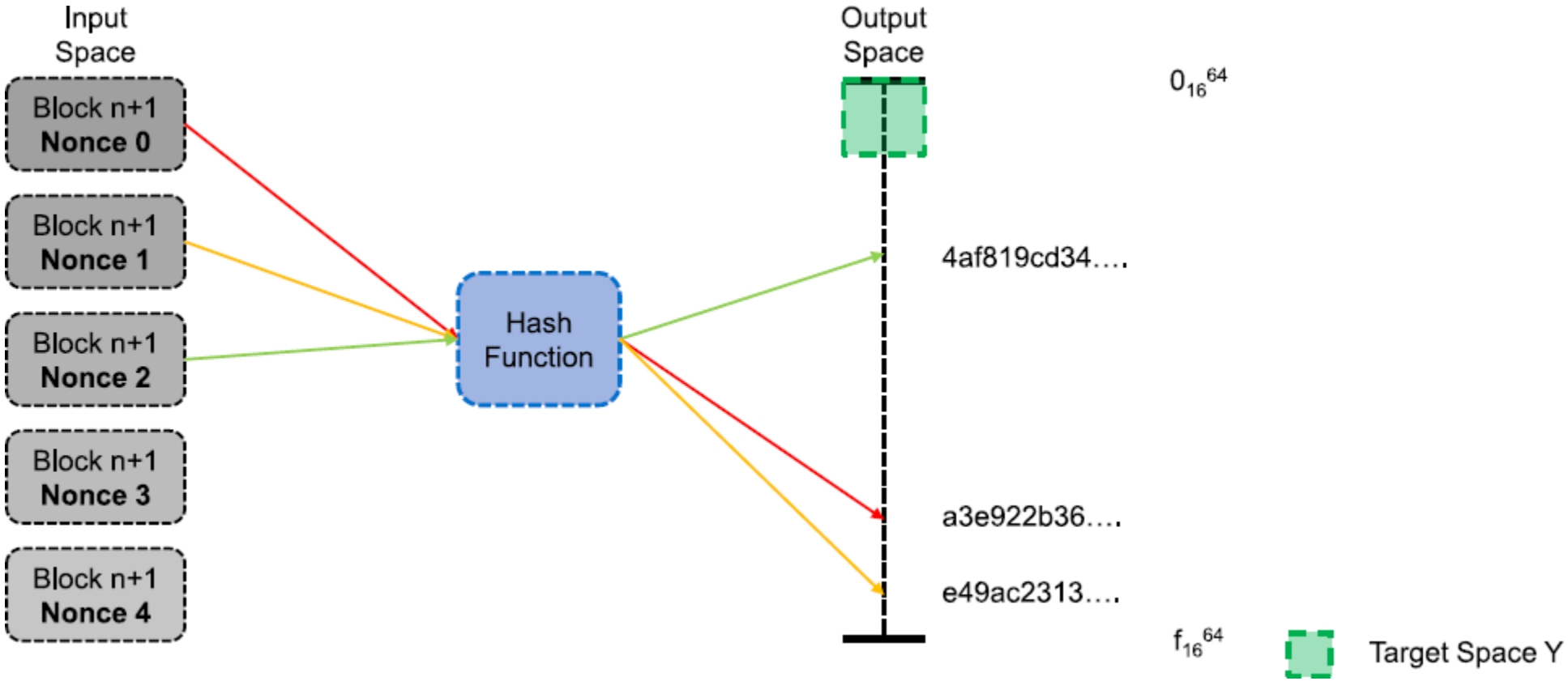
Idea: We use the search puzzle introduced in the chapter about cryptographic foundations. The header of the hash has to be included in Y. Bitcoin uses double SHA-256. (sha256(sha256(block)))



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Search puzzle

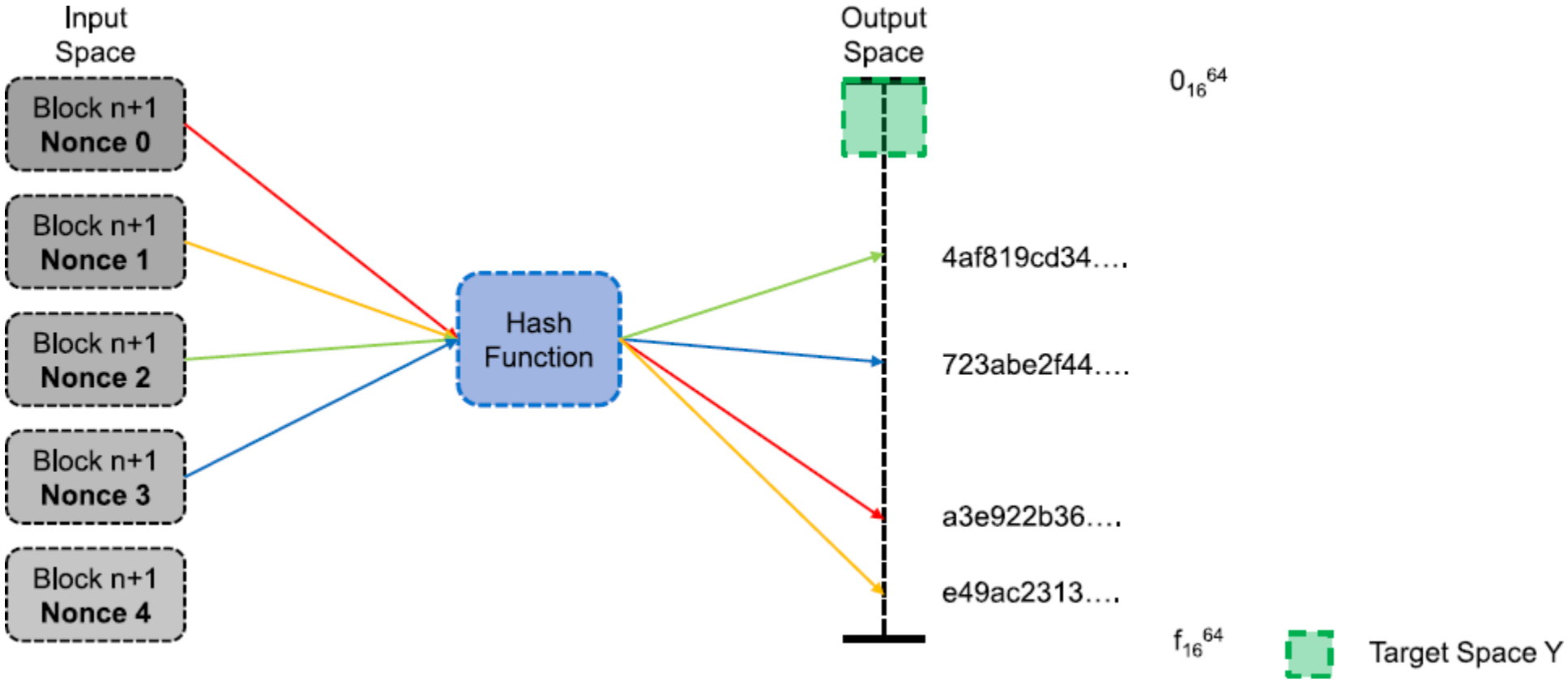
Idea: We use the search puzzle introduced in the chapter about cryptographic foundations. The header of the hash has to be included in Y. Bitcoin uses double SHA-256. (sha256(sha256(block)))



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Search puzzle

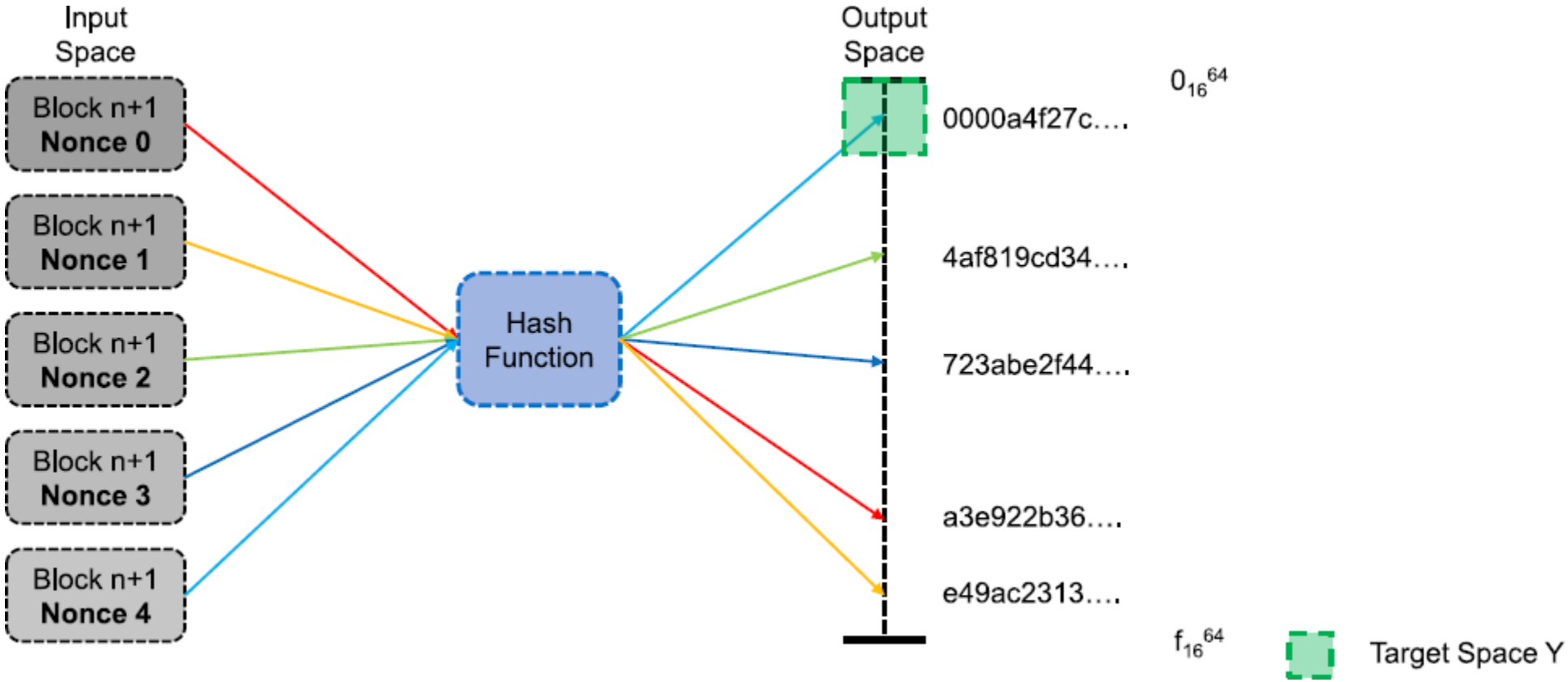
Idea: We use the search puzzle introduced in the chapter about cryptographic foundations. The header of the hash has to be included in Y. Bitcoin uses double SHA-256. (sha256(sha256(block)))



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Search puzzle

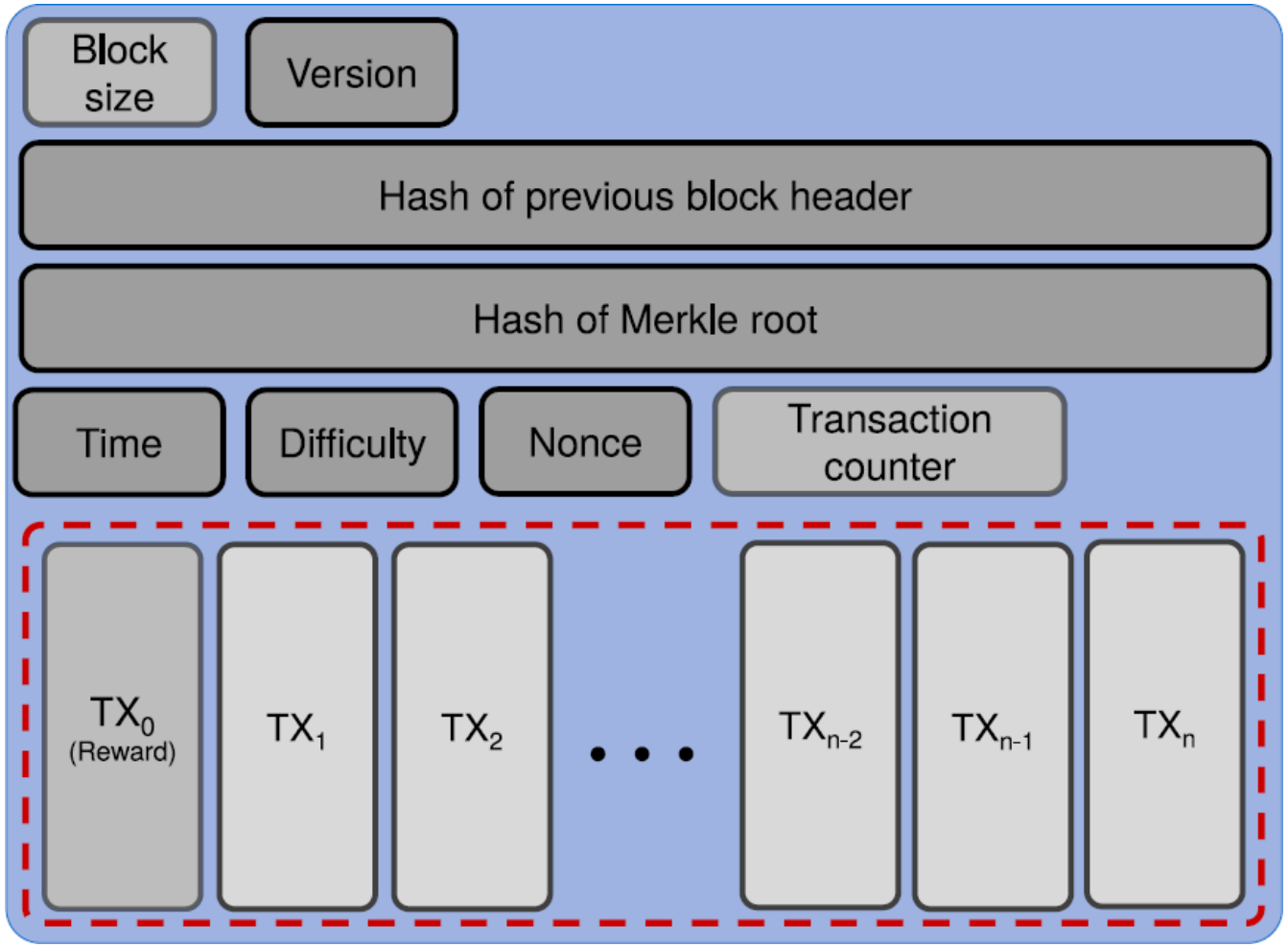
Idea: We use the search puzzle introduced in the chapter about cryptographic foundations. The header of the hash has to be included in Y. Bitcoin uses double SHA-256. (sha256(sha256(block)))



U. Gallersdörfer, P. Holl, and F. Matthes, "Blockchain-based Systems Engineering – Lecture Slides," Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Block structure

Note: Due to different mempools and reward addresses, miners will work on different puzzles.



U. Gallersdörfer, P. Holl, and F. Matthes, “Blockchain-based Systems Engineering – Lecture Slides,” Oct 2019. [Online]. Available: https://github.com/sebischair/bbse

Difficulty selection

The difficulty of the search puzzle is adjusted to keep the average time between blocks constant. This is either done dynamically or at fixed intervals.

In Bitcoin the target time between blocks is ten minutes and the difficulty is adjusted every 2016 blocks, which is roughly two weeks. The longest chain is considered to be the chain with the highest accumulated difficulty.

Difficulty selection

Why is the block time kept constant?

- If it is too slow, transactions take longer to be confirmed and network capacity decreases
- If it is too fast, blocks will be empty, more forks and orphaned blocks will occur

In Bitcoin, the difficulty is adjusted as follows:

- Measure how long the last 2016 blocks took to get mined (T)
- Calculate the factor of speed (F := 2w/T)
- The difficulty gets increased (F > 1) or decreased (F < 1)
- The maximum increase is 4, the maximum decrease is 0.25
- This process is done every 2016 blocks

Incentives

In Bitcoin, there are two incentives for mining.

Transaction fees: Every transaction has a fee attached, which is the difference between the sum of inputs and the sum of outputs. For example, a transaction spending inputs worth 2.5BTC to two outputs, one worth 1BTC and the second 1.2BTC would in effect be offering 0.3BTC worth of fees to incentivize miners into including it in a block. Due to the limited block size, transaction fees are used to bid for space in blocks with low amounts of fees per byte in the transaction incurring significantly longer confirmation times.

Mining reward: For each block, the miner is entitled to include a "coinbase" transaction that creates a certain amount of bitcoins (currently 6.25BTC per block) from nothing and sends it to an address of
their choice. The amount of bitcoins awarded in this matter is halved at regular intervals. As at some point this halving will be rounded down to zero, the total number of bitcoins is capped to a predefined amount (21 million).

Attacks

For PoW, the main attack scenario is the 51% attack, where a miner controls more than 50% of the hash power in the network. If this happens, this miner will over time always be able to create the longest chain, while maintaining their advantage in hash power. Due to this, they can:

- Censor transactions
- Reliably double spend with any number of confirmations
- Prevent other miners from getting rewards

However, keeping up such an attack is costly and requires investments in updated hardware if the advantage is to be maintained, as well as incurring high energy costs.

If the attack becomes known, the value of the target blockchain's currency will likely decrease, making it harder to recoup these costs. There are also other more subtle attacks, such as selfish mining.

PROOF OF STAKE

Proof of Stake (PoS)

While PoW implementations are usually similar, with PoS there are more differences. At its most basic, participants lock up a certain amount of the blockchain's base currency and thereby become eligible to be chosen as validators that create blocks.

There are two main approaches:

- **Chain based PoS:** An algorithm regularly (e.g. every 10s) pseudo-randomly selects a validator and assigns them the right to create the next block, which points to a previous block. Over time most blocks converge into a growing longest chain.
- **Byzantine Fault Tolerance (BFT)-style PoS:** Validators are chosen randomly to propose a block. Then, in a multiround process, every validator votes for which one should become canonical. At then end validators permanently agree which block is part of the chain. *The key difference is that consensus on a block can come within one block, and does not depend on the length or size of the chain after it.*

Incentives

Validators validate transactions before including them in blocks and are rewarded with the transaction fees for those transactions.

To ensure good behaviour, mechanisms may be implement that forfeit a validator's stake when fraudulent behaviour from the validator is detected. To make this work, the stake amount needs to be higher than the possible transaction fees gained.

Stake is time locked for some duration after the eligibility period ends to enable punishment for bad behaviour.

"Nothing at stake"

In early PoS approaches, no penalties for misbehaving validators were considered.

- Validators could do whatever they want
- When multiple chain heads exist, it's most lucrative to put a block on each
- The chain could never reach consensus

In chain based PoS, this can be mitigated by e.g. proof of misbehaviour based penalties to staked funds, either when validators add blocks to multiple chains or when they add them to the "wrong" one.

Approaches to solve this also exist in BFT-style PoS.

Attacks

Other attacks also exist on PoS based systems:

- **51% attack:** If someone controls 51% of the coins on a blockchain, but seems more difficult to achieve.
- **Stake grinding:** In some PoS systems, it was possible for validators to perform computations while staking or building blocks etc. to increase the chances they will be selected in the future.

IOTA – A BLOCKCHAIN FOR THE IOT

Issues of Existing Blockchains

- Bandwidth
- Storage
- Transaction fees
- Block size
- PoW energy consumption
- Required computing power not compatible with low energy IoT devices

IoT Requirements

- Low resource consumption
- Interoperability
- Large number of nano-transactions
- Data integrity

IOTA in a Nutshell

- Ledger of things
- Blockchain without blocks, also not really a chain
- Bundles all transaction into a directed acyclic graph (DAG)

The Tangle

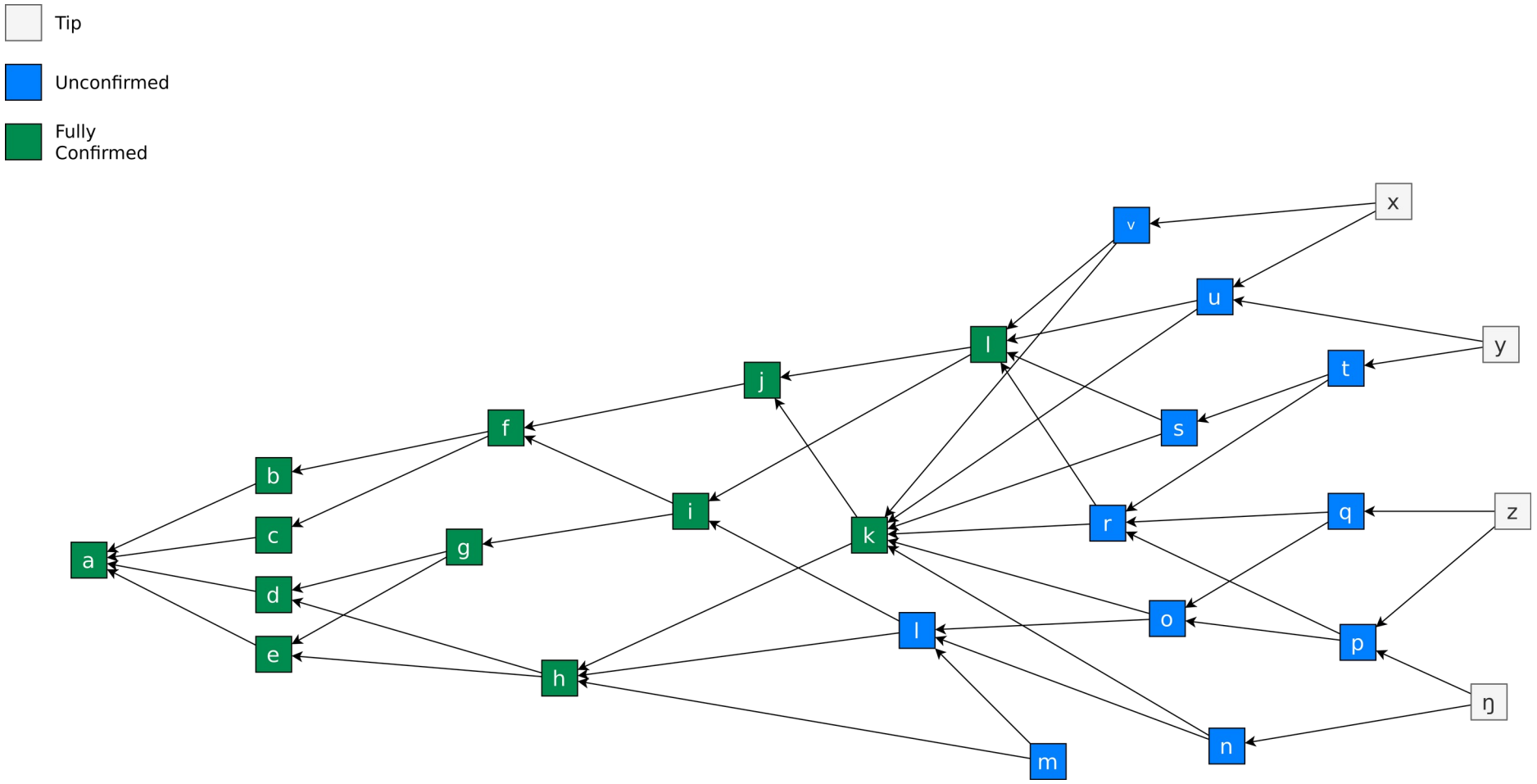


Image recreated based on https://github.com/noneymous/iota-consensus-presentation

How to make a Transaction?

1. Sign the transaction inputs with your private keys.
2. Tip selection based on a random walk - select two tips (they use Markov Chain Monte Carlo (MCMC))
3. PoW – Add a small PoW to your transaction so that your transaction is accepted by the network (spam protection + sybil resistance).

The Tangle – Adding Transactions

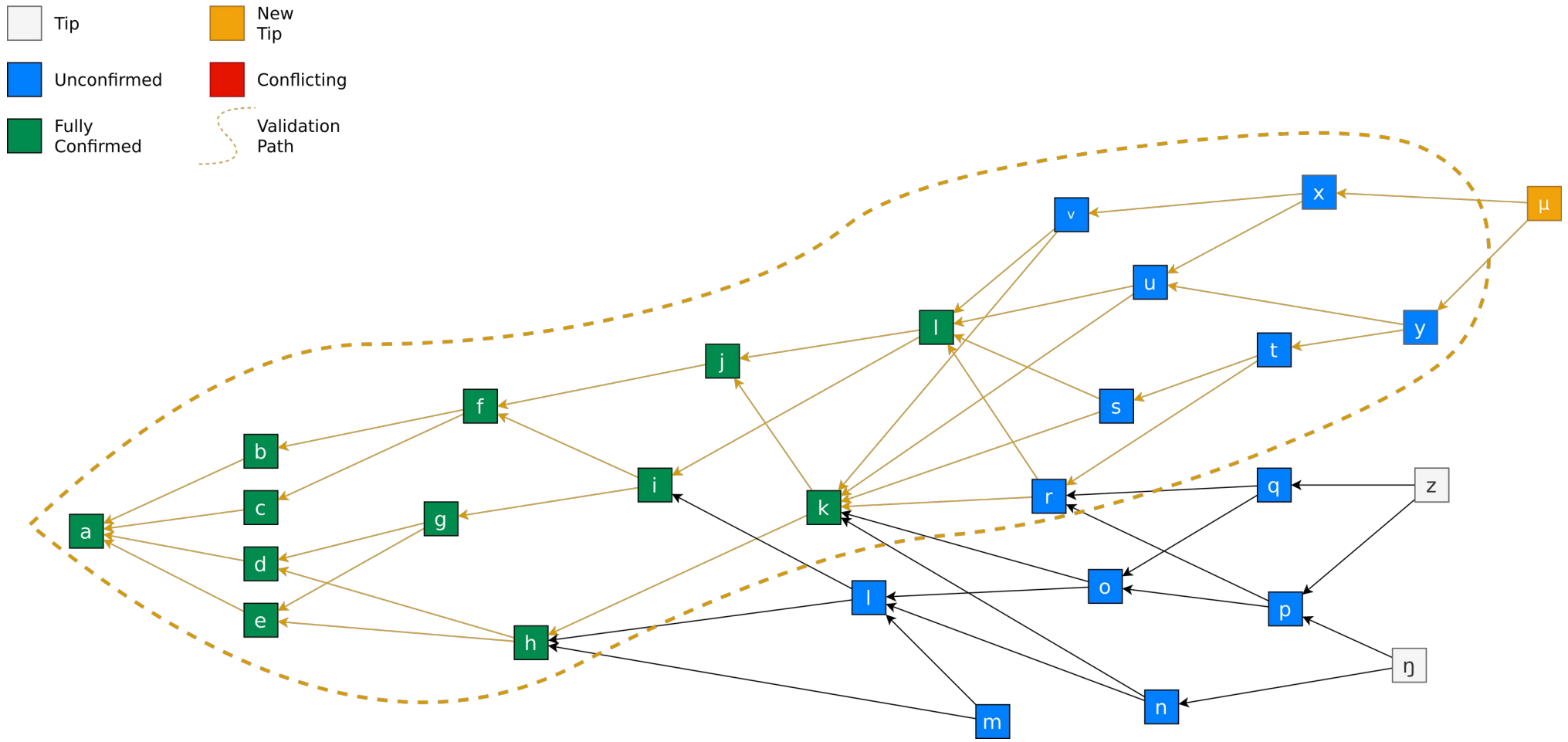


Image recreated based on https://github.com/noneymous/iota-consensus-presentation

The Tangle – Adding Transactions

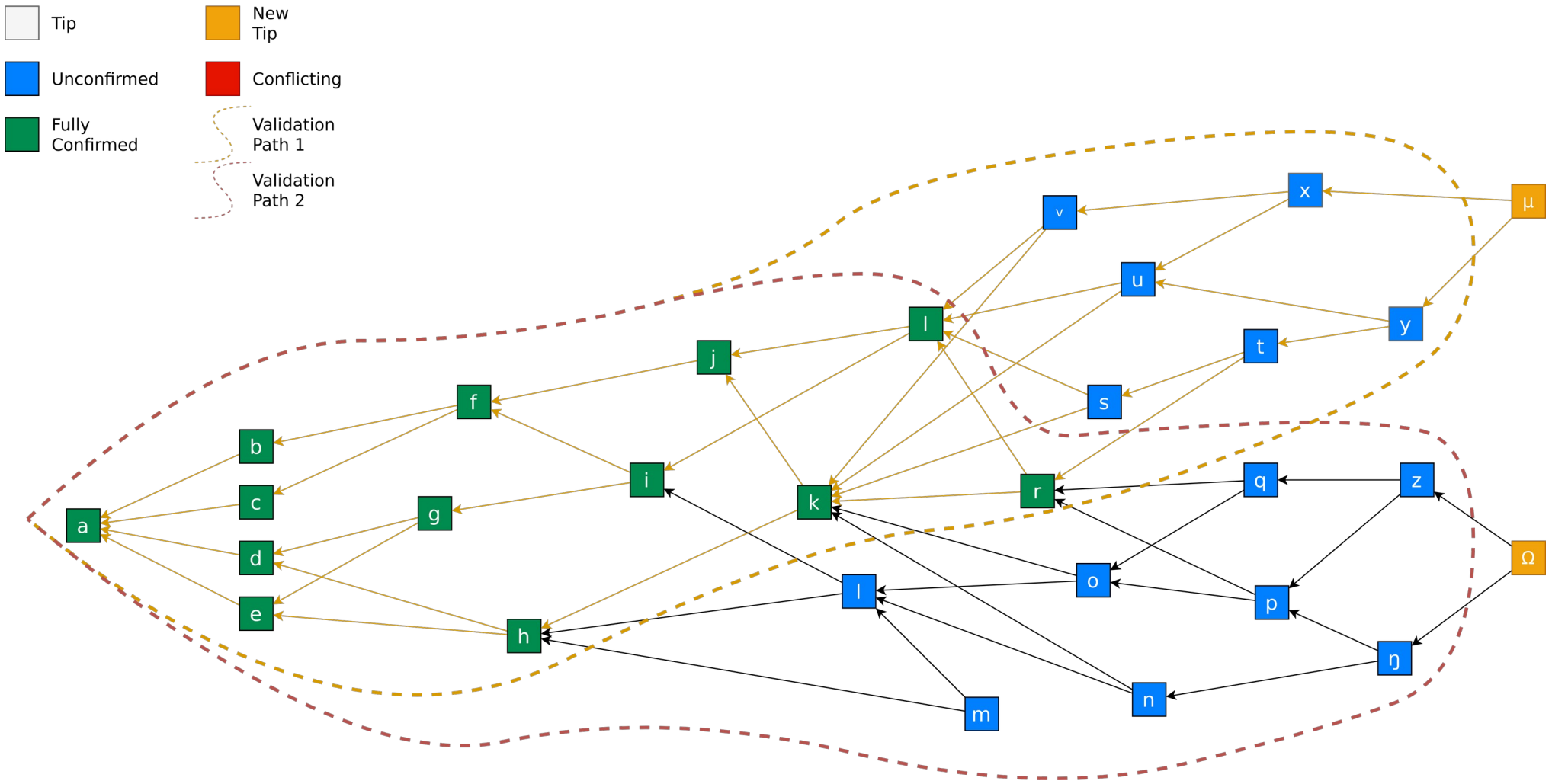
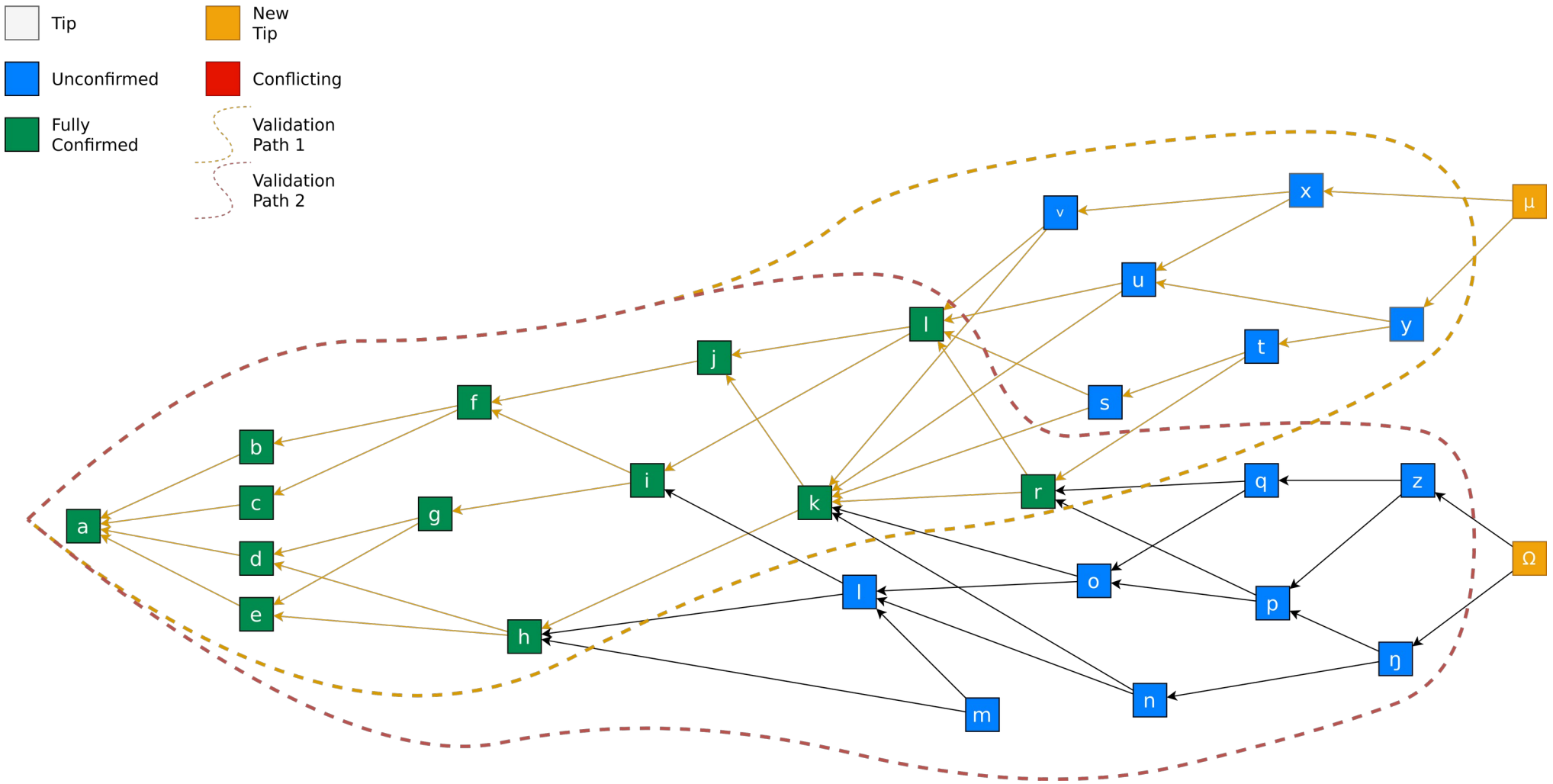


Image recreated based on https://github.com/noneymous/iota-consensus-presentation

The Tangle – Adding Transactions



→ Transactions validated and confirmed by all of the current tips are considered fully confirmed.

Image recreated based on https://github.com/noneymous/iota-consensus-presentation

The Tangle – Double Spending

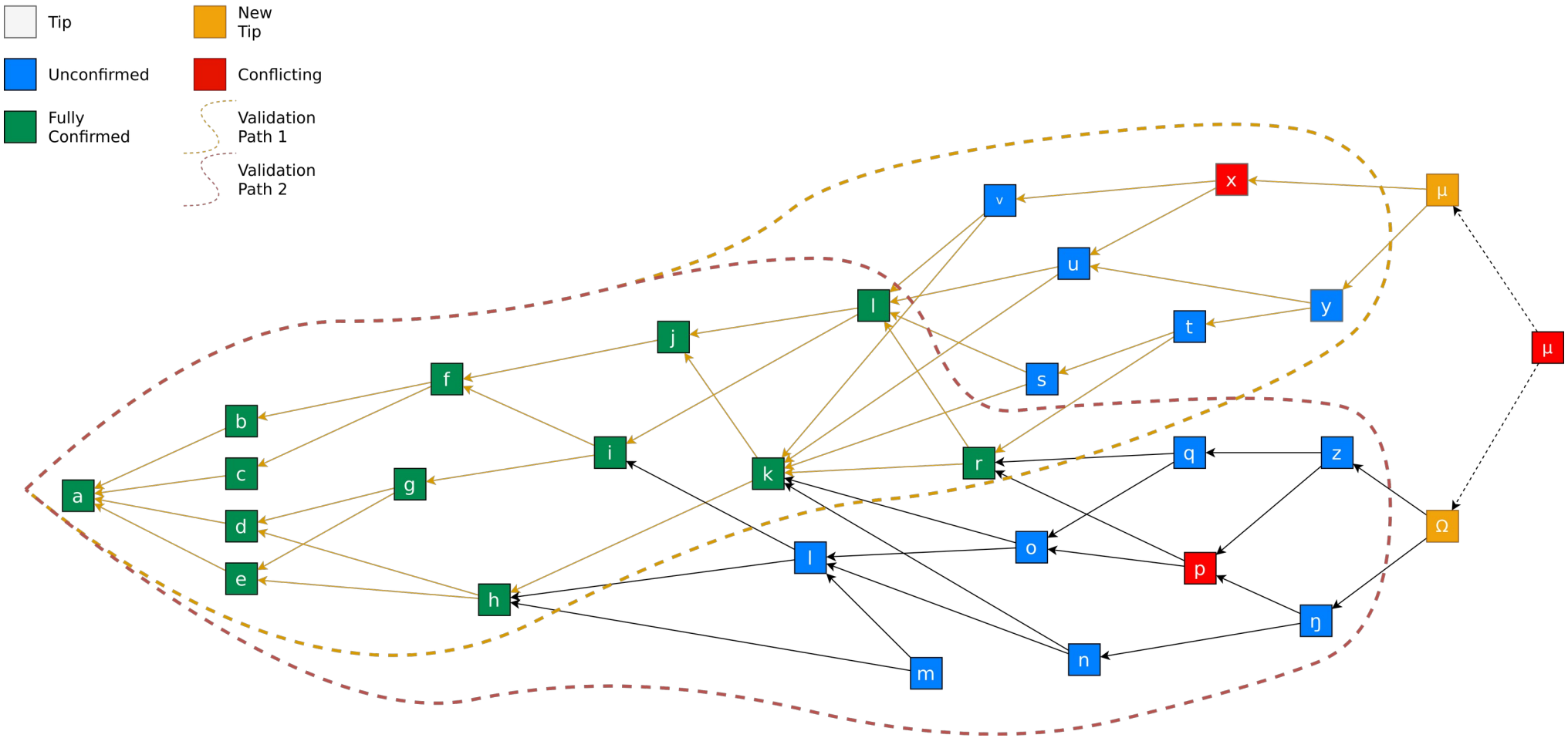


Image recreated based on https://github.com/noneymous/iota-consensus-presentation

Advantages of IOTA

- Scalable
- Claims quantum proofness
- No fees
- Lightweight
- Offline transactions

The Tangle – Offline Transactions

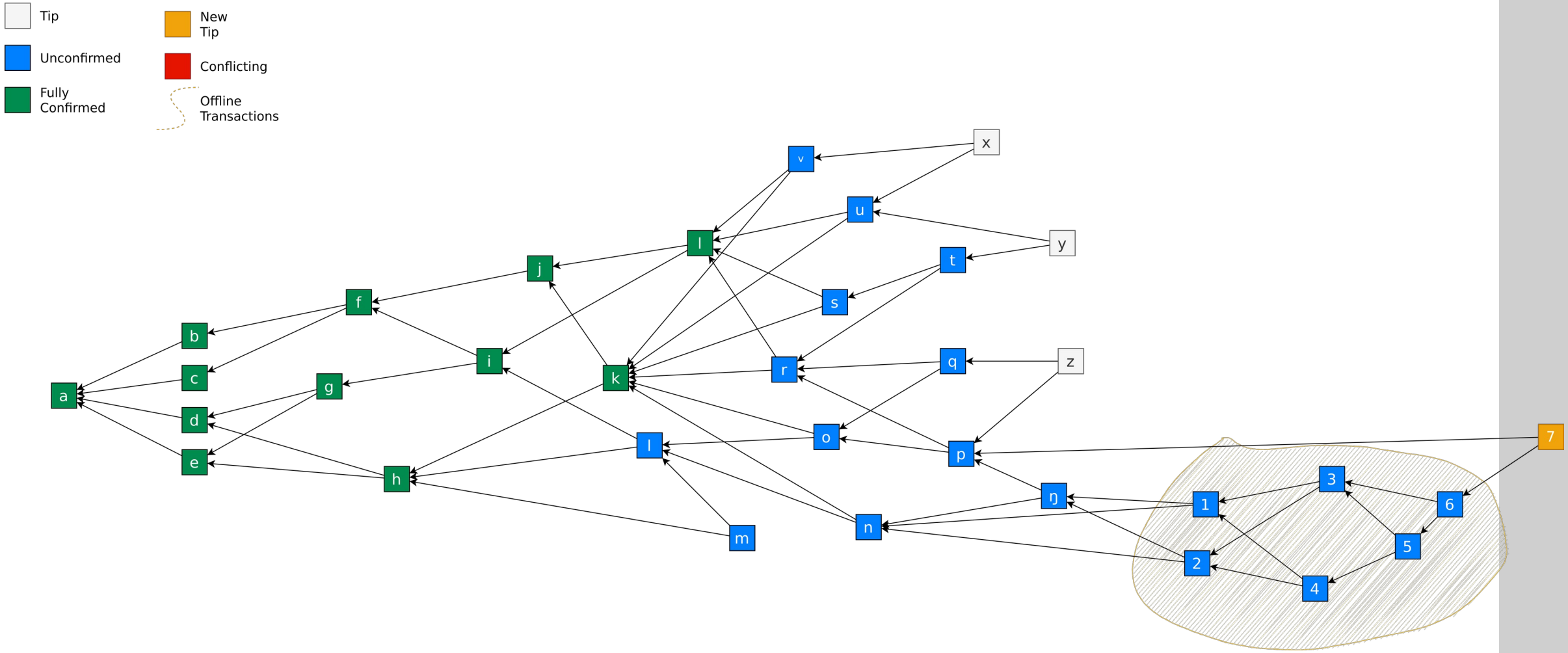


Image recreated based on https://github.com/noneymous/iota-consensus-presentation

Drawbacks

- Centralized: Coordinator nodes run by the IOTA foundation
- Unconventional structure for Blockchain => fresh Vulnerabilities

Conclusion

- Blockchain without blocks and without a chain
- Focused on IoT applications
- Directed acyclic graph
- Tangle
- Consensus = Small PoW + Tip Selection Algorithm
- No transaction fees

Further Resources on IOTA

- The Tangle – Link
- Equilibria in the Tangle – Link
- The first glance at the simulation of the Tangle: discrete model – Link
- Extracting Tangle Properties in Continuous Time via Large-Scale Simulations – Link
- Improving the Anonymity of the IOTA Cryptocurrency – Link
- Probability of Being Left Behind and Probability of Becoming Permanent Tip – Link
- Quasi-Analytic Parasite Chain Absorbtion Probabilities in the Tangle – Link
- On the timestamps in the tangle – Link

Questions?