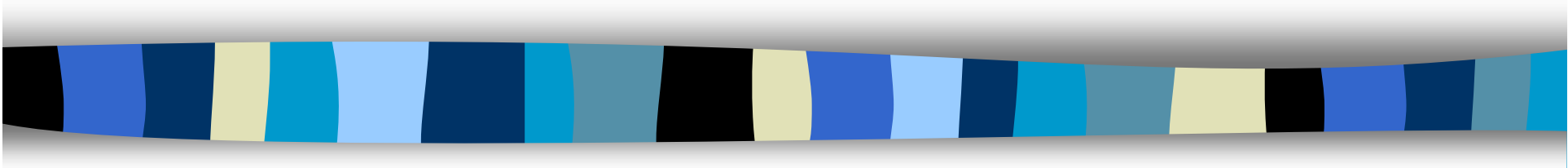


6CCS3AIN AI Reasoning & Decision-Making: Consensus Mechanisms

Week 9 — Part C — Consensus in Blockchain



Peter McBurney
Department of Informatics
King's College London
London

peter.mcburney@kcl.ac.uk

Week 9
30 November 2020



Introduction

- This is Part C of the lecture for Week 9.
- The topic for this week is consensus mechanisms, which are an important part of automated decision-making in decentralized and distributed systems.
- In this lecture, we look at the type of mechanisms used to reach consensus in Blockchain systems, as in the Bitcoin Cryptocurrency.
 - We begin with a few words about Bitcoin and Blockchain.
- The material in this Part C is intended to show you an actual and very significant application of these ideas about automated decision-making in decentralized networks of autonomous agents.
 - The details of the Bitcoin Blockchain are not examinable in this course.

Bitcoin (BTC)

- Bitcoin was conceived in 2008, and was deployed in 2009.
 - Inventor was "Satoshi Nakamoto" (a pseudonym)
 - This is the world's first decentralized electronic currency
- Key problem of an electronic currency:
 - How to ensure there is no double-spending of currency
 - Prior solution: Have a central node in charge of allocation
 - Weakness: Central node may be corrupt or vulnerable to attack
- Bitcoin's solution:
 - Have the allocations of currency witnessed and approved by all nodes in the network.
 - And chaining of data makes it impossible to change past data surreptitiously.





Blockchain

- The underlying technology was called Blockchain
- There was no new invention, just a clever combination of previous CS technologies
 - Peer-to-peer (P2P) network
 - Consensus Protocol for automated collective decision-making
 - Proof-of-Work to enable appropriate economic incentives
 - Cryptography for authentication of participants (PKI – Public Key Infrastructure)
- Basic atoms: Transactions (TXs) shifting allocation of some currency from one node to another node
- Transactions grouped into blocks, then blocks chained together
 - A new block about every 10 minutes
- Now part of Distributed Ledger Technologies (DLT).



Economic incentives

- The designers of the system want to incentivize participation
 - Processing transactions takes some work
- So the system rewards uploading blocks
 - The first node to do so correctly is rewarded with some BTC
- But the network is open and pseudonymous
 - Participation is open and is under a BTC address (no real names or ID info)
 - So a malicious node could join multiple times
 - Possibility of a Sybil attack (multiple-identity attack)
 - Sybil is not to be confused with sibyl (an oracle)
- So the network needs to discourage this
 - To successfully upload a new block, the node has to do some additional processing work
 - This involves solving a math puzzle - “Proof-of-Work” (PoW)
 - The puzzle is hard to solve but easy to check if correct.



Four parts of decentralized consensus

- A Independent verification of each transaction, by every full node
- B Independent aggregation of those transactions into new blocks by mining nodes
together with demonstrated computation through a Proof-of-Work algorithm
- C Independent verification of the new blocks by every node and assembly into a chain
- D Independent selection, by every node, of the chain with the most cumulative computation demonstrated through Proof-of-Work.



The high-level process #1

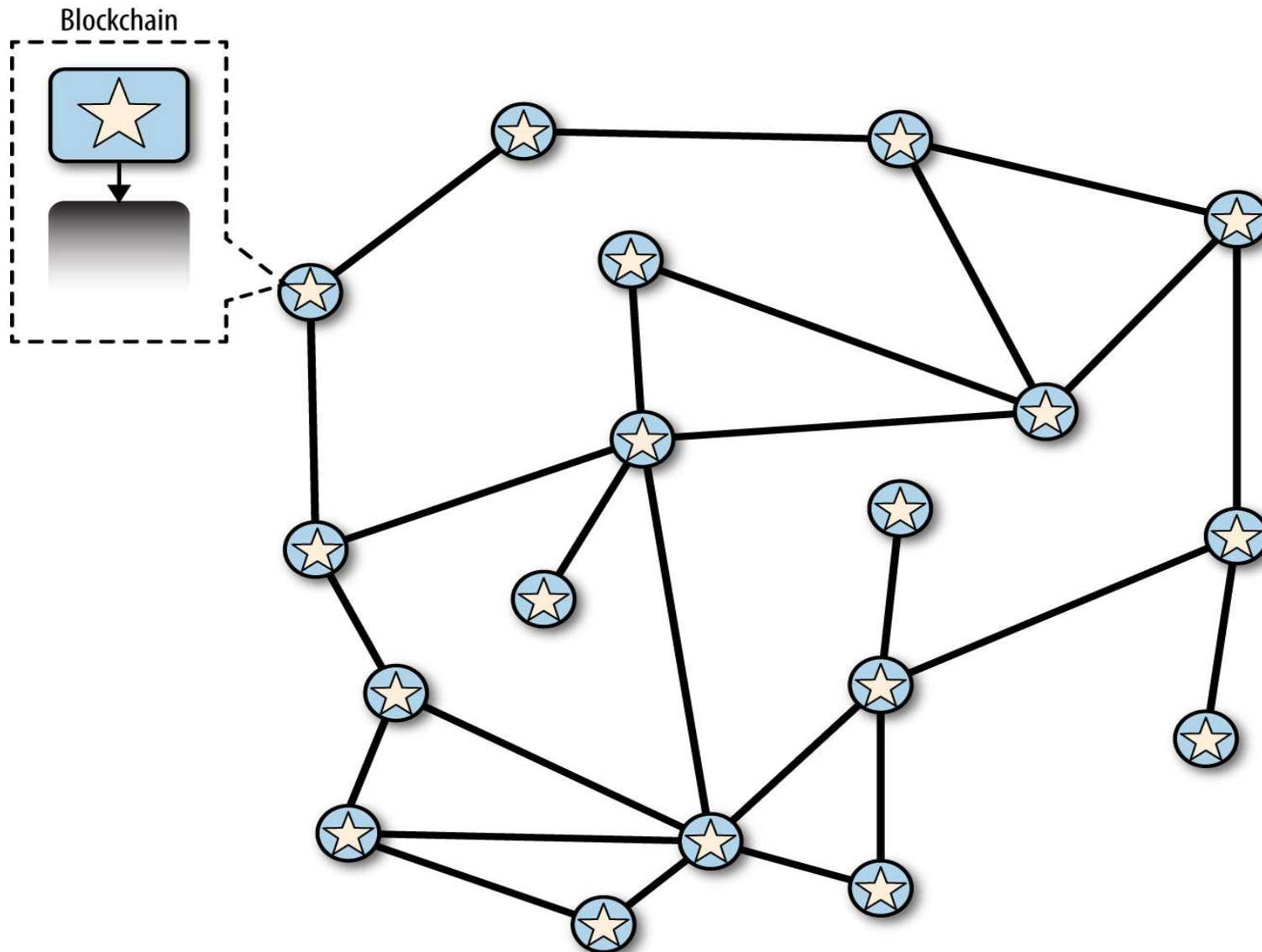
- We have a P2P network
 - Currently about 11,300 nodes
 - No one is in charge
 - All full nodes have the same software
- A node issues a potential transaction (TX)
 - Sent to all nodes in its neighbourhood (those it knows about)
- Each receiving node checks the potential TX for syntax and validity
- If validated, the receiving node sends the TX to all its neighbours, and so on
 - If the same TX is received more than once, subsequent receipts are ignored.



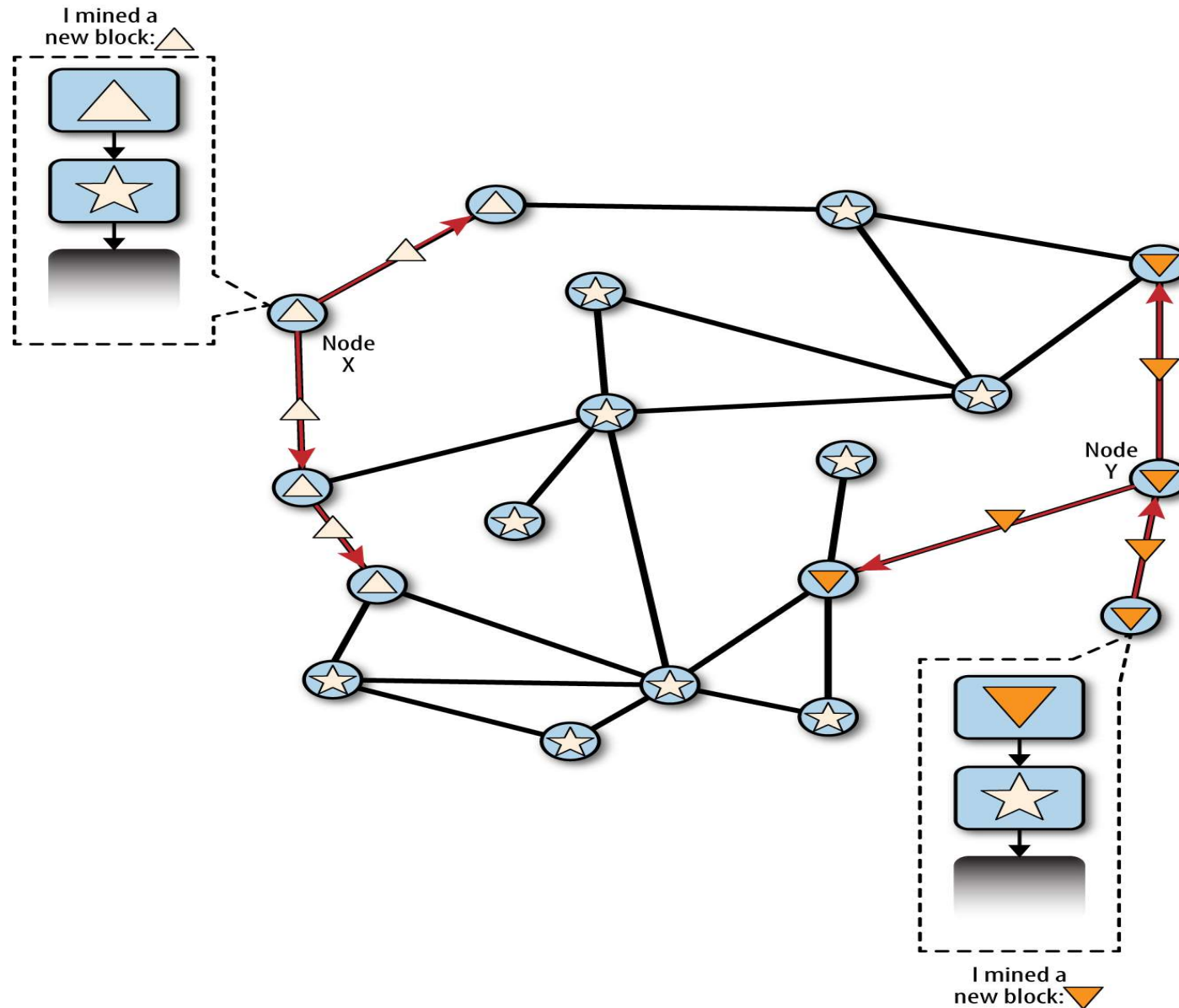
The high-level process #2

- Full nodes try to assemble a new block
 - They concatenate new TXs together
- Full nodes try to solve the math puzzle (by doing work)
 - If they solve it, they add the solution (the Proof-of-Work) to the block header
- They then send the candidate block to their neighbours
- Each receiving node validates the candidate block and adds it to its copy of the existing chain, then sends to its neighbours
- When faced with competing blocks on the chain, decides to accept the chain with the greatest cumulative proof of work.
 - This is the decision to adopt a particular state (or colour)
 - This is called The Longest-Chain-Rule.

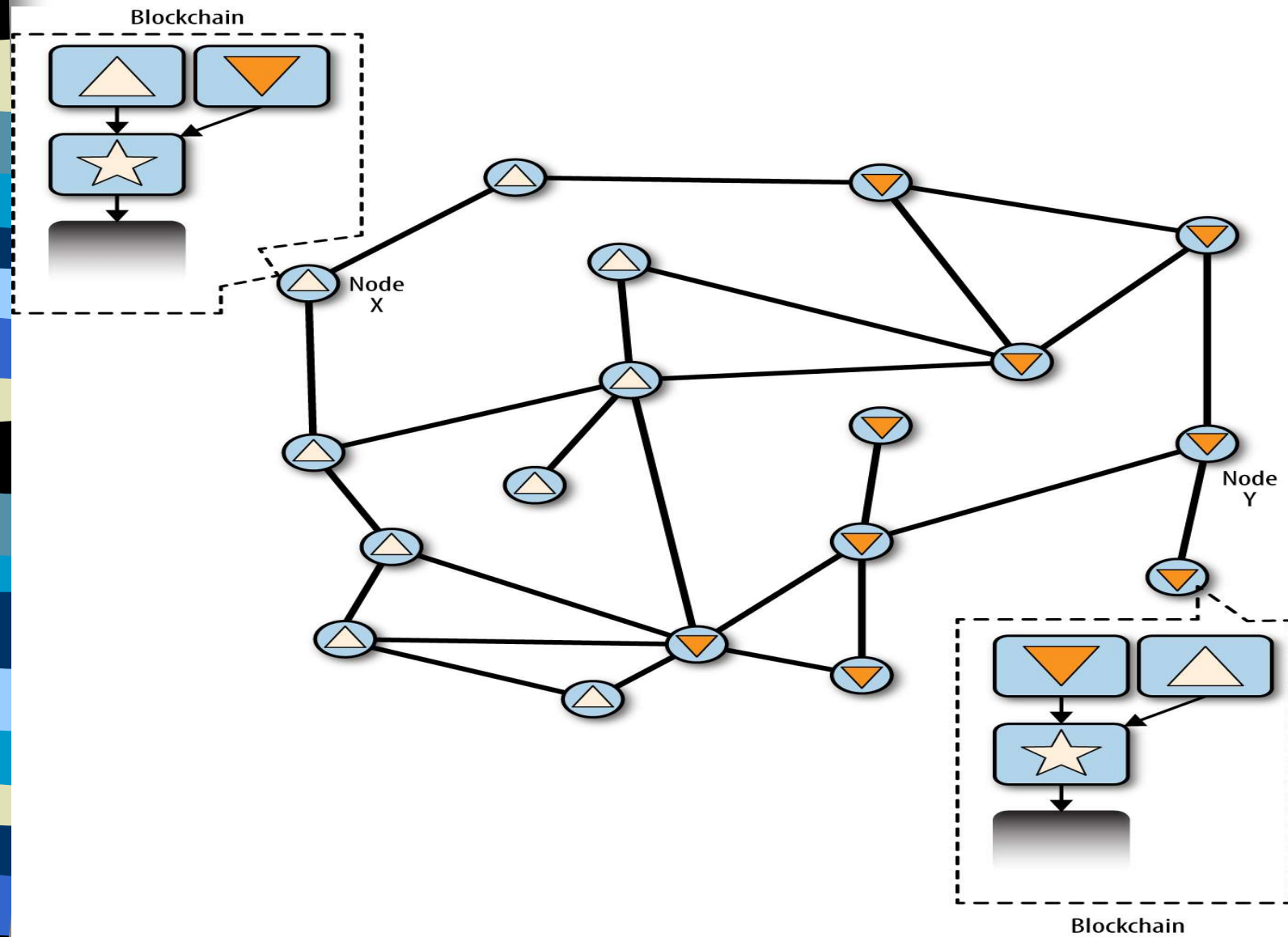
Blockchain assumes a peer-to-peer (P2P) network
No one is in control.



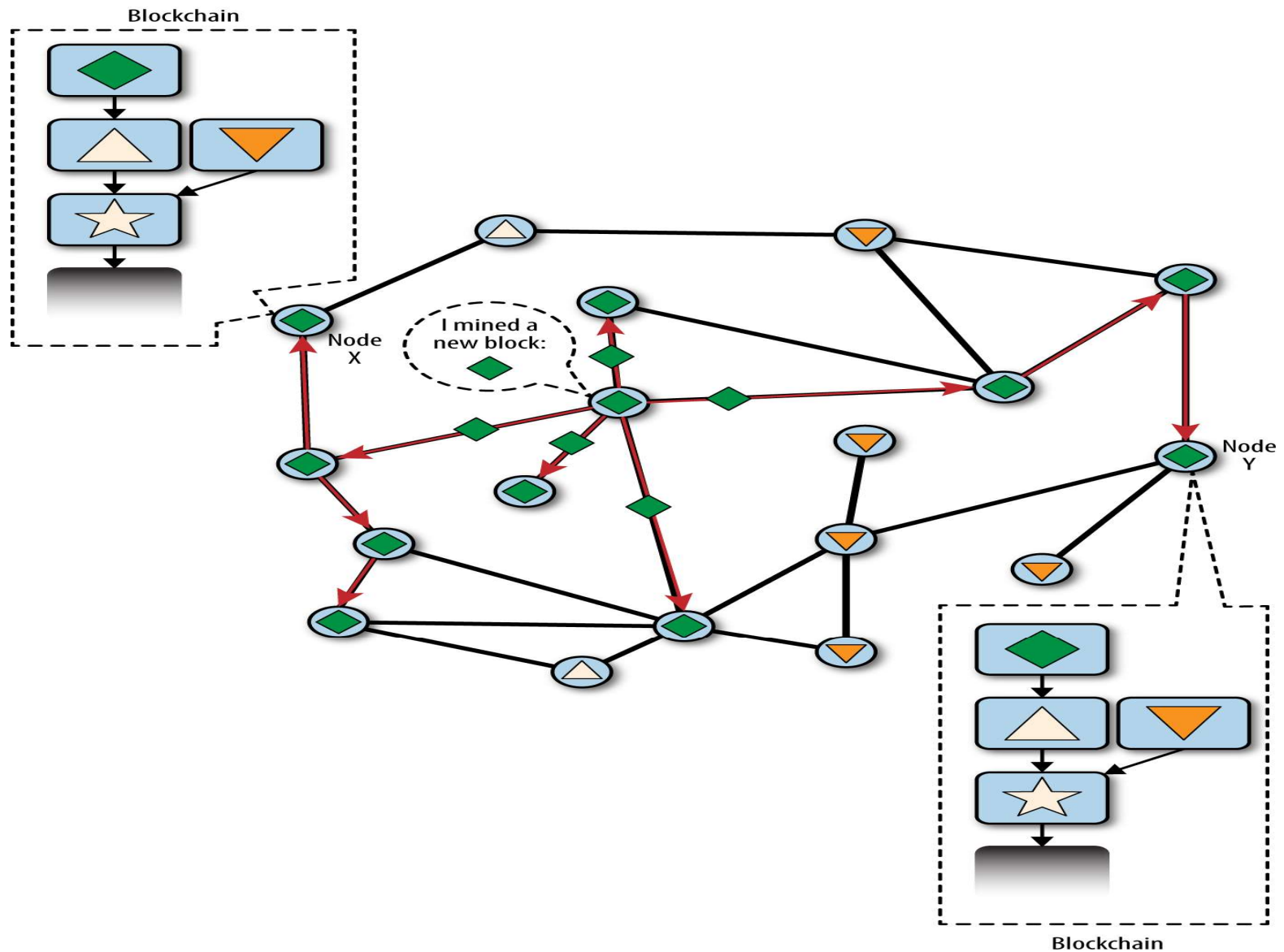
Nodes mine blocks and propagate them locally



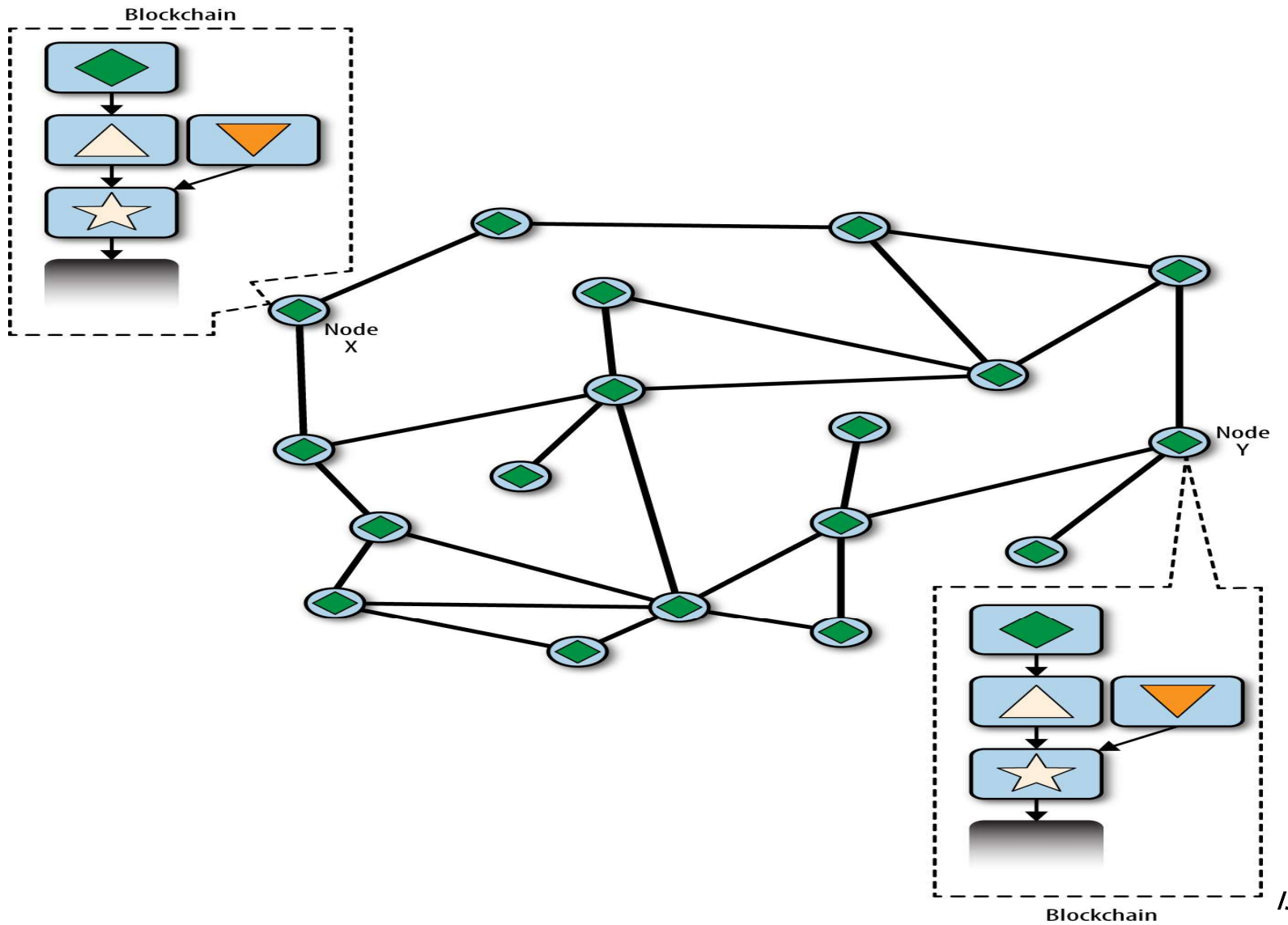
Competing new blocks from different miners



Which chain is “longer” (contains more work)?



Eventually consensus is achieved





Resolving differences

- Because of network delays, the longest chain in one part of the network may be different to that in another part of the network.
- Faced with competing blocks, each node adds the different blocks to the existing chain, and then chooses the chain with the greatest cumulative work.
- If the result is a tie, the node waits until another valid block is received.
- If the result is again a tie, the process continues until a clear answer emerges.
- Most often, the conflict (called a temporary fork) is resolved within 2 or 3 blocks (ie, 20-30 minutes)
 - The worst case involved 24 blocks (but due to a bug in a software upgrade).



Summary I

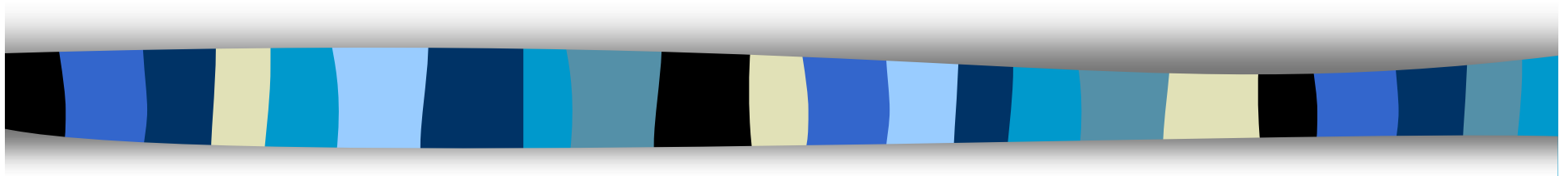
- The Bitcoin Blockchain uses a consensus protocol for all nodes in the network to agree on addition of each new block to the existing blockchain.
- The decisions are made independently and automatically by each node, each one using the exact same algorithm.
- The decision by each node is made on basis of information received from their neighbours (new TXs and new blocks).
- The system works well if all nodes are acting correctly and in good faith.



Summary 2

- The system does not work well if nodes have bugs or are acting maliciously.
- But this should be only temporary
- If a malicious node has inserted a false block this will eventually be superseded by other valid blocks.
- To maintain the false block in the chain over an extended period, the malicious node needs to keep mining new blocks on top of the false block.
 - But this is costly for the malicious node, since it requires the node to do ongoing work in the PoW process.

Thankyou!





Appendix

- The next slides are for your interest only. They are not part of 6CCS3AIN and are not examinable.



Mining new bitcoin

- New bitcoin are created during the creation of each block at a fixed and diminishing rate, approx. every 10 minutes.
- Every 210,000 blocks (ca. four years), the currency issuance rate is decreased by 50%
 - 2009-2012: 50 new bitcoin earned per block
 - November 2012: Reward became 25 new bitcoin per block
 - July 2016: Reward became 12.5 bitcoin per block
 - 8 April 2020: Reward became 6.25 bitcoin at block 630,000
 - ca. 2137: 1 satoshi per block (block 6,720,000) (99% of all BTC)
 - ca. 2140: After 6.93 million blocks a total of almost 2,099,999,997,690,000 satoshis (almost 21 million bitcoin).
- After that, payment to miners will only be via transaction fees.

Reward for mining is new Bitcoin





Mining problem

- Proof-of-Work is designed to create a hurdle to mining
 - Otherwise, nodes would spin-up multiple sock-puppet nodes to win the reward
 - A form of Sybil attack
- The problems get harder over time
 - To ensure that a new block is created about every 10 minutes.
- Problem: Find the hash a specified object with a nonce parameter which is less than sum pre-specified total.
 - Problem designed to be hard to do and easy to check.
 - Can only be solved by trial and error.



Four parts of decentralized consensus

- A Independent verification of each transaction, by every full node
- B Independent aggregation of those transactions into new blocks by mining nodes
together with demonstrated computation through a Proof-of-Work algorithm
- C Independent verification of the new blocks by every node and assembly into a chain
- D Independent selection, by every node, of the chain with the most cumulative computation demonstrated through Proof-of-Work.



A: Independent verification of transactions

Each node checks against the following list of criteria:

- The transaction's syntax and data structure is correct.
- Neither lists of inputs or outputs are empty.
- The transaction size in bytes is less than MAX_BLOCK_SIZE.
- Each output value, as well as the total, is within the allowed range of values
- None of the inputs have hash=0, N=-1 (coinbase transactions should not be relayed)
- nLocktime is equal to INT_MAX, or nLocktime and nSequence values are satisfied according to MedianTimePast.
- The transaction size in bytes is greater than or equal to 100.
- The number of signature operations (SIGOPS) contained in the transaction is less than the signature operation limit.
- The unlocking script can only push numbers on the stack, and the locking script must match isStandard forms.
- A matching transaction in the pool, or in a block in the main branch, must exist.
- For each input, if the referenced output exists in any other transaction in the pool, the transaction is rejected.
- For each input, look in the main branch and the transaction pool to find the referenced output transaction. If the output transaction is missing for any input, this will be an orphan transaction. Add to the orphan transactions pool, if a matching transaction is not already in the pool.
- For each input, if the referenced output transaction is a coinbase output, it must have at least COINBASE_MATURITY confirmations.
- For each input, the referenced output must exist and cannot already be spent.
- Using the referenced output transactions to get input values, check that each input value, as well as the sum, are in the allowed range of values (less than 21m coins, more than 0).
- Reject if the sum of input values is less than sum of output values.
- Reject if transaction fee would be too low (minRelayTxFee) to get into an empty block.
- The unlocking scripts for each input must validate against the corresponding output locking scripts.



Four parts of decentralized consensus: C & D

- C Independent verification of the new blocks by every node and assembly into a chain
- D Independent selection, by every node, of the chain with the most cumulative computation demonstrated through Proof-of-Work.
 - We can reference blocks by their height (about 660,000), or the hash of their header.
 - Block height may not be unique (if there is a fork).
 - Block hash is not stored within the block
 - It is calculated by each node as the block is received.



Validating a new block

Criteria for validation include:

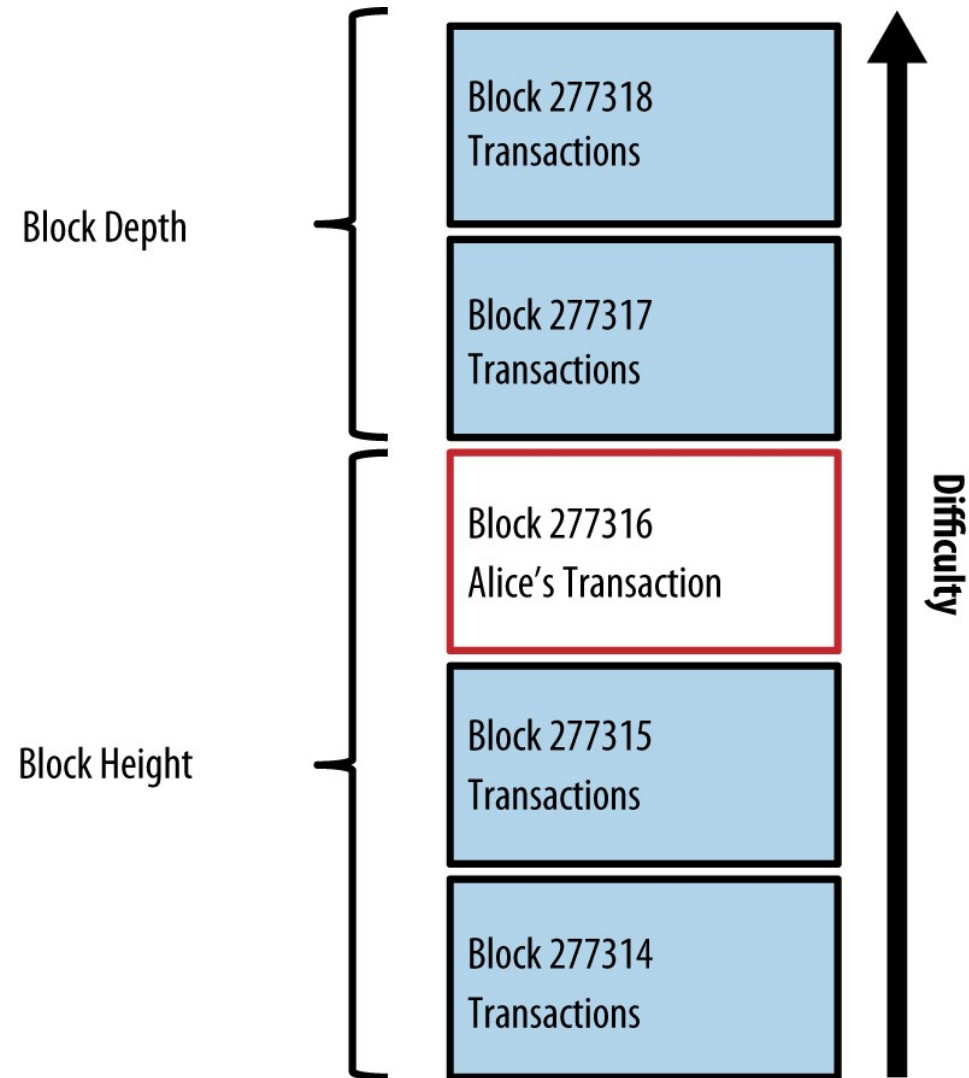
- The block data structure is syntactically valid
- The block header hash is less than the target (enforces Proof-of-Work)
- The block timestamp is less than two hours in the future (allowing for time errors)
- The block size is within acceptable limits
- The first transaction (and only the first) is a coinbase transaction
- All transactions within the block are valid using the transaction checklist for Independent Verification of Transactions.



How do nodes decide between competing blocks?

- Nodes keep three collections of blocks
 - Those on the main blockchain
 - Those that form branches off the main blockchain
 - Orphan blocks – those without a parent block
- The main chain is the chain with the most cumulative difficulty associated with it
 - Usually the chain with the most blocks
 - If two chains are equal length, then the main chain is the one with most PoW
 - Forks are usually resolved within 1-3 blocks
- 10 minutes for each block time is a compromise between
 - Fast confirmation times & the probability of a fork.
- As computer power increases, the Bitcoin blockchain automatically adjusts other parameters to ensure average block time is around 10 minutes.

Block height currently is about 659,050 (on 2020-11-28)



<https://blockchain.info/q/getblockcount>