23. hroug
godišnja konferencija

Novel blockchain consensus algorithm

doc. dr. sc. **Martin Žagar**
dr. sc. **Branko Mihaljević**
Andrej Šarić
HUJAK, RIT Croatia

About the Authors

- dr.sc. **Branko Mihaljević**
  - PhD in CS, mr. sc., DSM
  - Educator, Software Architect, Tech Consultant, Entrepreneur: 18+ years in Education (RIT Croatia, FER, Algebra, VERN, 20+ workshops…) and in Business (CEO, CTO, Project Management, 20+ large projects…)
  - Interests: Web Application Architectures, Programming Languages (Java), Artificial Immune Systems, E-learning, Traffic Problems, Distributed Data Architectures, and Blockchain
  - President (& founder) of HUJAK, JCP Associate Member…
- doc. dr. sc. **Martin Žagar**
  - PhD in CS, EMBA (Cotrugli BS), mr. sc. in Eco-engineering, DSM
  - Educator (RIT Croatia, FER, VVG), Scientist (Academy awarded), Researcher (FP7 and H2020 projects)
  - IEEE, HiPEAC, and AASCIT member
  - Interests: Multimedia, Computer Architectures, Telemedicine, e-Health, m-Health, IT Systems, High performance computing, Blockchain
- **Andrej Šarić**
  - Entrepreneur, Educator (RIT Croatia)
  - Interests: Smart Cities, Mobile Development, Web Development, Blockchain

Before We Start

- What do you need to know before you start with blockchain development?
  - Algorithms and Data Structures
  - Cryptography
  - Digital Signatures
  - Application Development
  - Web Dapp

Agenda

- About Blockchain
- Consensus Algorithms
- Hashgraph
- Gossip
- RRG
- HashNET
- Conclusion and Future Work

About Blockchain

- Chain of blocks
- Contains information
- Distributed ledger
- Once data is recorded it cannot be changed
- Each block contains
  - Data
  - Hash
  - Hash of the previous block
- Hash is like a fingerprint
- If data inside the block changes so will the hash

Blockchain Characteristics

- Decentralized and Distributed
- No single point of failure
- Immutable
- Almost real-time synchronization
- Unstoppable

Consensus in General

- In a centralized system a central administrator has the authority to maintain and update the database
- Blockchains operate as decentralized and self-regulating systems
- How to agree on a consensus on the status of the ledger

Various Consensus Protocols (or "Proofs")

- Current High-Profile Consensus Algorithms:
- Proof of Work (**PoW**) – Bitcoin, Ethereum, Bitcoin Cash, Litecoin…
- Proof of Stake (**PoS**) – new Ethereum, Cardano
- **Ripple** Protocol Consensus Algorithm – Ripple
- **Stellar** Consensus Protocol – Stellar
- Delegated Proof of Stake (**dPoS**) – EOS
- **Delegated BFT** – NEO
- Proof of Importance (**PoI**) – NEM
- Proof of Luck (**PoL**)
- Proof of eXercise (**PoX**)

Proof of Work

- The Proof of work concept existed even before bitcoin
- Protocol with the main goal of deterring cyber-attacks such as a distributed denial-of-service attack (DDoS)
- Defines an expensive computer calculation (mining)
- Servers two purposes:
  - Verifying the legitimacy of a transaction
  - Creating new digital currencies by rewarding miners

Proof of Work

- Transactions bundled together into a block
- Miners need to verify that transactions within each block are legitimate
- To verify transactions miners solve a mathematical puzzle known as proof-of-work problem
- Reward is given to the first miner who solves each blocks problem
- Verified transactions are stored in the public blockchain

Proof of Stake

- Algorithm with the same purpose like the proof of work
- Suggested in 2011
- Coins are created at the beginning and their number is fixed
- Deterministic way to choose creator of a new block
- Depends on wealth of the creator (called stake)
- No block reward
- Miners take the transaction fees

Other High-profile Consensus Algorithms

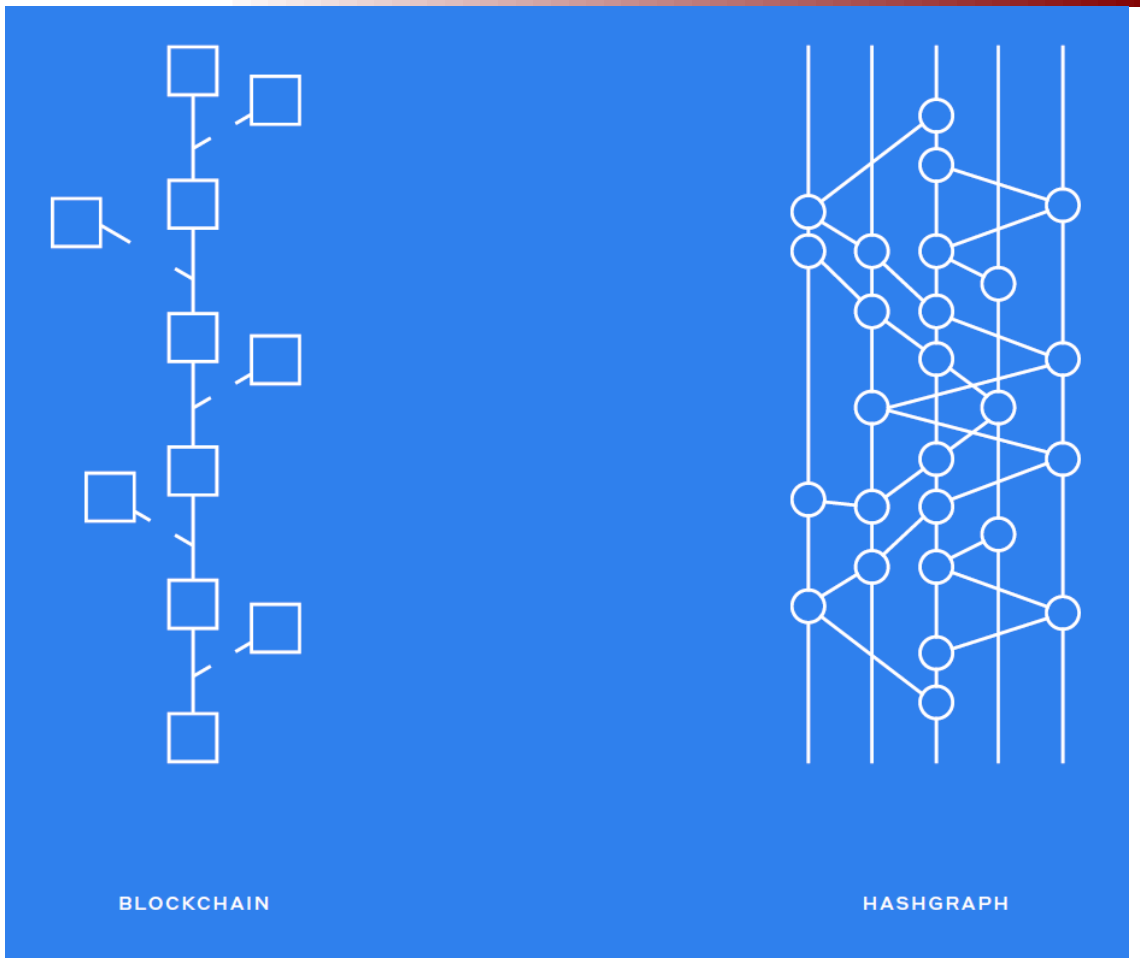
- Ripple
- Stellar
- Tangle

Consensus Algorithm Characteristics

- Not easy to compare because of different characteristics …

	Algorithm Name				
Property	PoW	PoS	PBFT	DPoS	Ripple
Energy Saving	No	Partial	Yes	Partial	Yes
Tolerated power of adversary	< 25% computing power	<51% stake	< 33.3% replicas	< 51% validators	<20% faulty nodes

Hashgraph



- Blockchain is like a tree that is continuously pruned as it grows – this pruning is necessary to keep the branches from growing out of control
- In hashgraph, rather than pruning new growth, it is woven back into the body

Hashgraph features (1)

- In both blockchain and hashgraph, any member can create a transaction, which will eventually be put into a container (the “block”), and will then spread throughout the community
- In blockchain, those containers are intended to form a single, long chain
- If two miners create two blocks at the same time, the community will eventually choose one to continue, and discard the other one
- In hashgraph, every container is used, and none are discarded
- All the branches continue to exist forever, and eventually grow back together into a single whole, in a more efficient way

Hashgraph features (2)

- Blockchain fails if the new containers arrive too quickly, because new branches sprout faster than they can be pruned
- That is why blockchain needs proof-of-work or some other mechanism to artificially slow down the growth
- In hashgraph, nothing is thrown away
- There is no harm in the structure growing quickly
- Every member can create transactions and containers whenever they want
- So it is very simple, and tends to be very fast

Hashgraph features (3)

- Because the hashgraph doesn’t require pruning and therefore is simpler, it allows more powerful mathematical guarantees, such as Byzantine agreement and fairness
- Blockchain is neither Byzantine nor fair
- Hashgraph is both Byzantine and fair

Gossip Protocols

- Gossip protocol is a communication protocol that is based on the way social networks disseminate information
- Features:
  - Periodic, pairwise, inter-process interactions
  - Information exchanged during these interactions is of bounded size
  - State of at least one agent changes to reflect the state of the other
  - Reliable communication is not assumed
  - Frequency of the interactions is low so the protocol costs are negligible
  - Randomness in the peer selection of nodes
  - Replication causes an implicit redundancy of the delivered information

Redundancy Reduced Gossip (RRG)

- Can achieve a considerably lower traffic load
- Compared to conventional push-based gossip protocols and conventional push-pull gossip protocols
- For the same probability of successful delivery
- With higher performance gains in networks with smaller delays

HashNET

- Idea: Combine Hashgraph features with Gossip protocol
- Additionally: Use Redundancy Reduced Gossip (RRG)

- Goals:
- Scalable
- Fast
- Fair
- Secure

HashNET

- Innovative consensus platform based on Hashgraph distributed consensus algorithm
- Novel solution to computational and communicational difficulties of maintaining large-size public distributed ledgers
- Provides significant reduction of computational and communication resources needed to operate and maintain the system
- Uses Redundancy Reduced Gossip (Improved RRG) protocol for information transfer on suitably designed network

HashNET

- Solves issues with scalability and energy consumption
- Scalability (200,000 transactions per second)
- Uses gossip protocol (RRG)
- Proof of Stake with masternodes
- Virtual voting
- Full node can be functional on a smartphone device

Building HashNET graph

- Nodes send out events to each other while gossiping
- Directed acyclic graph of connecting the nodes will grow
- Graph is called HashNET because it is connected by cryptographic hashes
- The entire graph is cryptographically secure since each event (vertex in the graph) contains the hashes of the events below it and it is digitally signed by the creator

Building HashNET graph

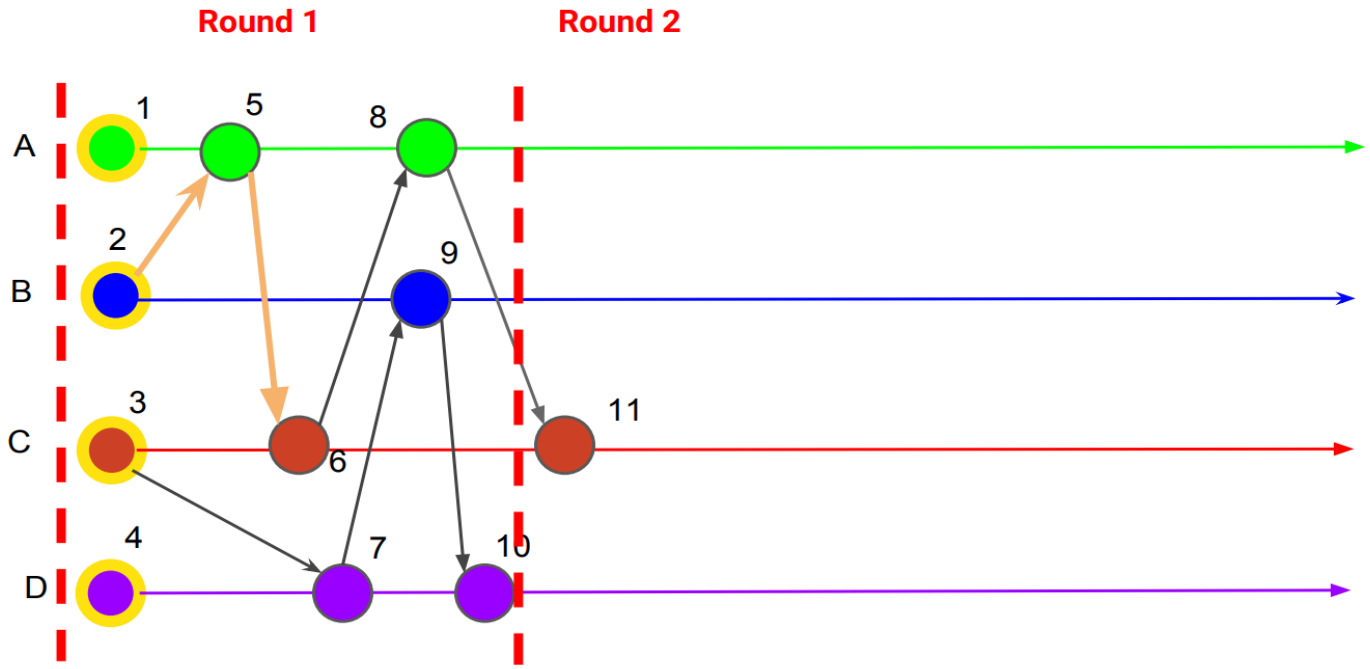
- Event – data structure created by some node and containing the two hashes of the preceding events
  - Parent event created by the same node ("self-parent") and the
  - Parent event created by some other node ("other-parent").
  - Creator of the transaction puts a timestamp to the event object at the creation time, and the event is thus digitally signed
- Events can have zero transactions either when
  - Node receives a sync event (HashNET difference)
  - When the node has just been spawned

Why HashNET

- Changes underlying data structure (DAG – directed acyclic graph)
- P2P network of masternodes
- Directed acyclic network structure
- Virtual voting – Masternodes can calculate how other masternodes will vote
- Vertices are connect with hash pointers
- Provides security
- Data is spread through gossip

Direct Path

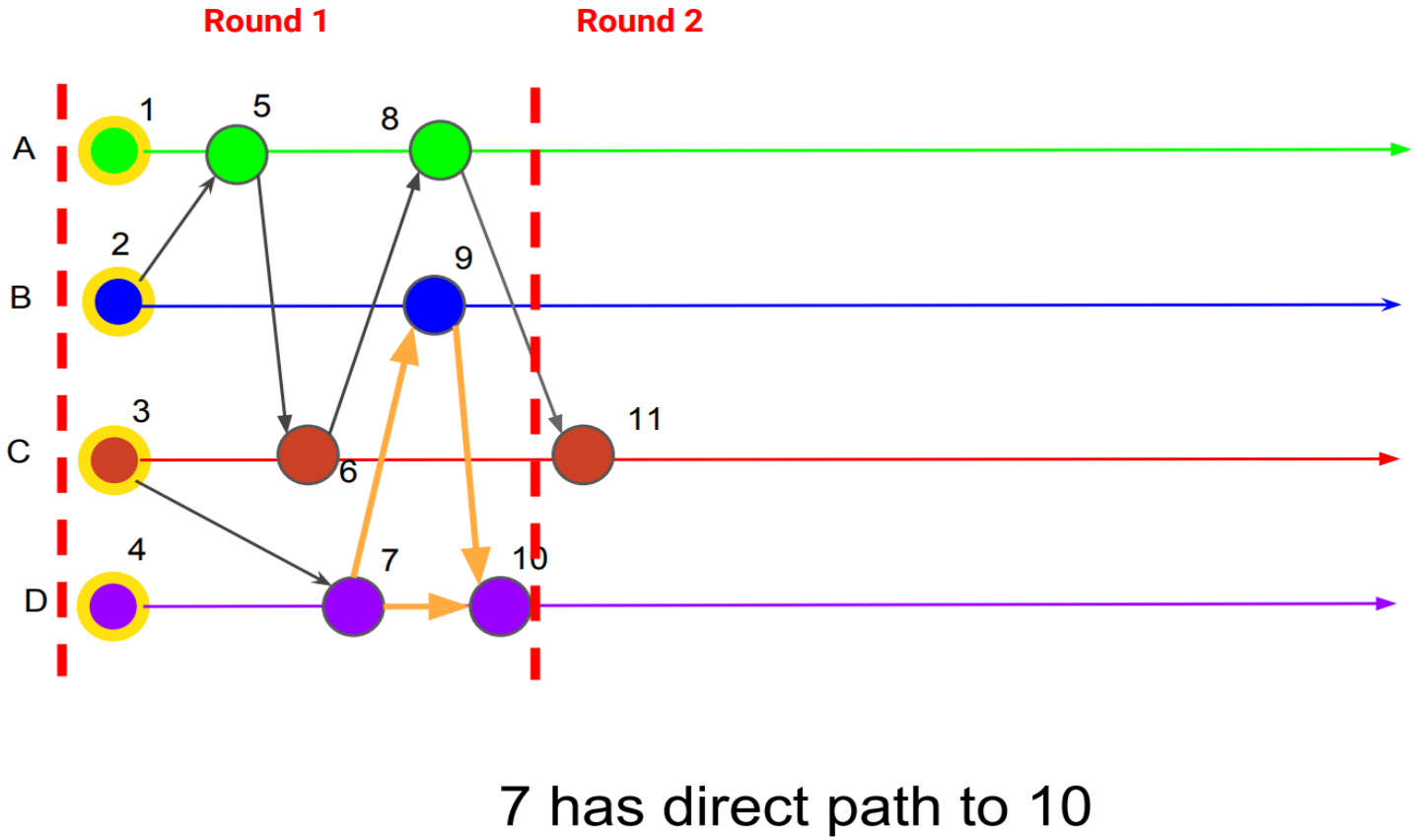
- The direct path exists if there exists any graph path in the directed acyclic graph
- Here we see a direct path from Event 2 to Event 6



2 has direct path to 6

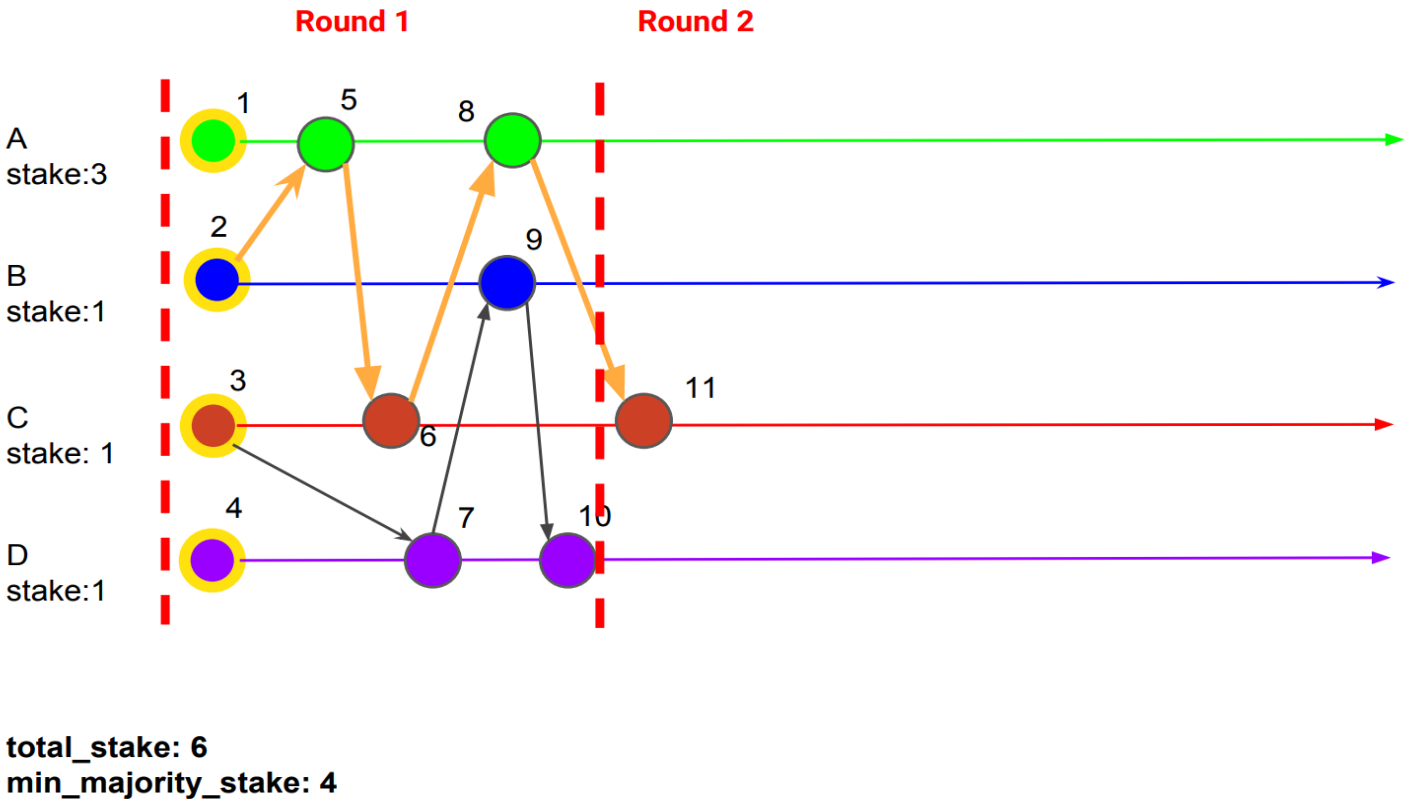
Direct Path

- Here is also a direct path from Event 7 to Event 10 – in this case, there are two different paths
- An event X strongly sees event Y if they are connected by multiple directed paths passing through a hyper-majority of nodes



Hyper Path

- Path from 2 to 11 goes through nodes A, B and C – the sum of the stakes for all nodes which it has been through is 5
- path_stake = A stake + B_stake + C_stake = 5
- IF path_stake >= min_majority_stake: path is Hyper path



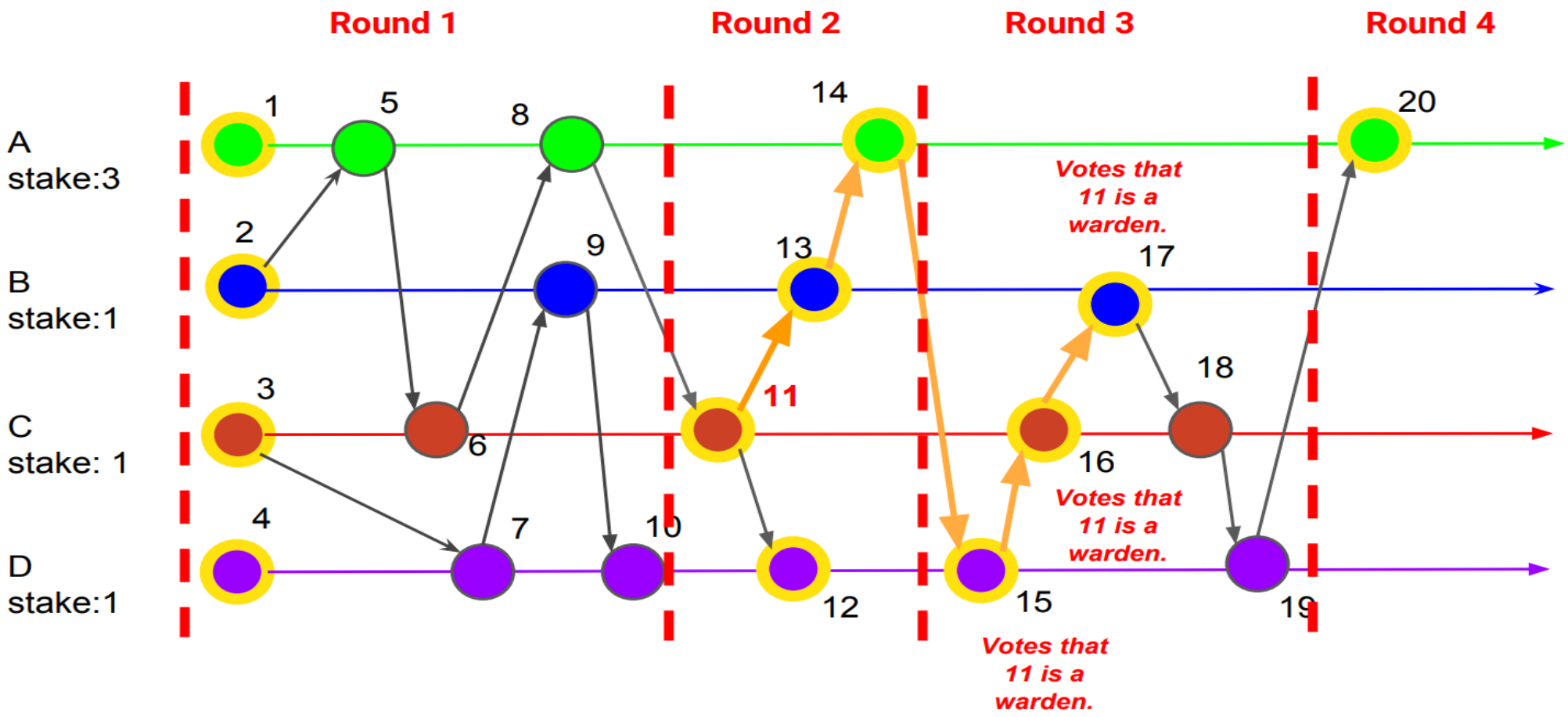
Achieving Consensus in a Network

- We could be:
  - Asking all nodes simple yes/no questions on whether an event X came before event Y
  - Running separate Byzantine agreement protocols which would require O(N log N) for such questions
- But much faster approach is to define some events as sentinels, and some sentinels to be wardens

Achieving Consensus in a Network

- Sentinels
  - First event created by a node in each round
- Wardens
  - For a round R sentinel, every R+1 sentinel is voting is the sentinel a warden or not
  - If an R+1 sentinel has a Direct path to the R sentinel, it votes that the sentinel is a warden
- Once a round has the wardens decided for all of its sentinels, round is received and consensus timestamp can be determined

Achieving Consensus in a Network



Comparison of Energy Usage

Currency Name	Consensus Protocol	Energy use per transaction (in kWh)
Bitcoin	Proof of Work (SHA256)	1011
Ethereum	Proof of Work (Equihash)	76
VISA		0.00169
IOTA	Tangle	0.0017
EOS	Delegated Proof of Stake	0.0053
Tolar	HashNET	0.00243

Comparison of Time to Finality

Currency Name	Consensus Protocol	Calculated Time to Finality (in seconds)
Bitcoin	Proof of Work (SHA256)	3600
Ethereum	Proof of Work (Equihash)	90
VISA		6
IOTA	Tangle	90
EOS	Delegated Proof of Stake	180
Tolar	HashNET	3

Real-world Applications

- Medical records
- E-notary
- Collecting taxes
- Etc.

Future Work

- Oriented on resolving upcoming problems in post-quantum cryptography with the goal of avoiding issues such as elliptic curve signature scheme

Conclusion

- Superior performance of HashNET algorithm compared to other consensus algorithms
- HashNET improvements compared to others:
  - Network throughput increase
  - Significantly lower energy consumption
- Time to Finality for transaction is comparatively low which is making it possible for real-world practical uses

Thank You

- Questions?