NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Analysis of the XRP Ledger Consensus Protocol

Xu Xiufeng
Wu Bozhi

Outline

Part 1: Introduction

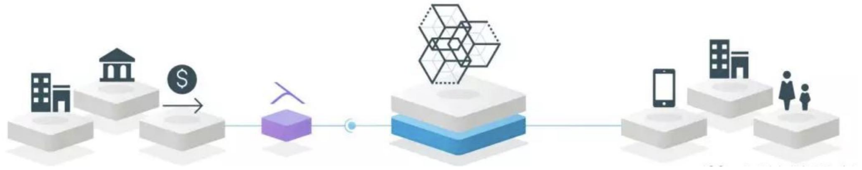
- ➢ Background
- ➢ Protocol definitions
- ➢ Protocol components

Part 2: Algorithm and Analysis

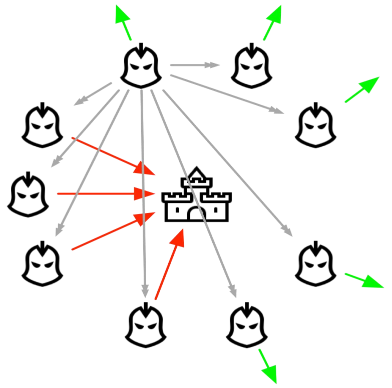
- ➢ Algorithm
- ➢ Analysis
- ✓ Safety
- ✓ liveness

Background

- The **XRP Ledger** is a distributed payment system enabling users to transfer value seamlessly around the world.

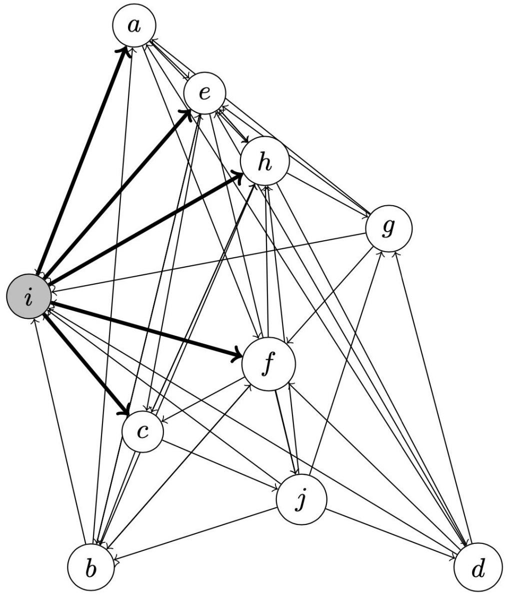


- Popular Consensus Algorithms:
  - PoW, PoS, DPoS, PBFT, ...
- XRP Ledger Consensus Protocol:
  - adopt Practical Byzantine Fault Tolerance
  - guarantee consistency with only partial agreement
  - **faster** consensus speed and guarantee finality of results.



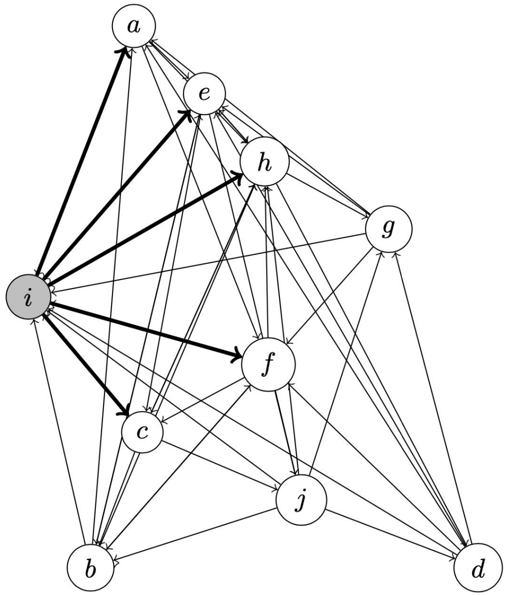
Basics of XRP LCP

- Let **Pi** be a node with unique identifier **i**
- Each **Pi** will maintain a **unique node list(UNL)**
- UNL represents a subset of the network which is trusted by Pi to not collude in an attempt to defraud the network.
  - individual node has complete discretion in selecting UNL
  - minimum overlap with other UNLs is necessary for consistency and liveness.
  - trust is not symmetric



Basics of XRP LCP

- Honest or correct nodes：
  - not crashed
  - behave exactly according to XRP LCP specification
- Byzantine nodes：
  - not responding to messages
  - sending incorrect messages
  - sending different messages to different parties

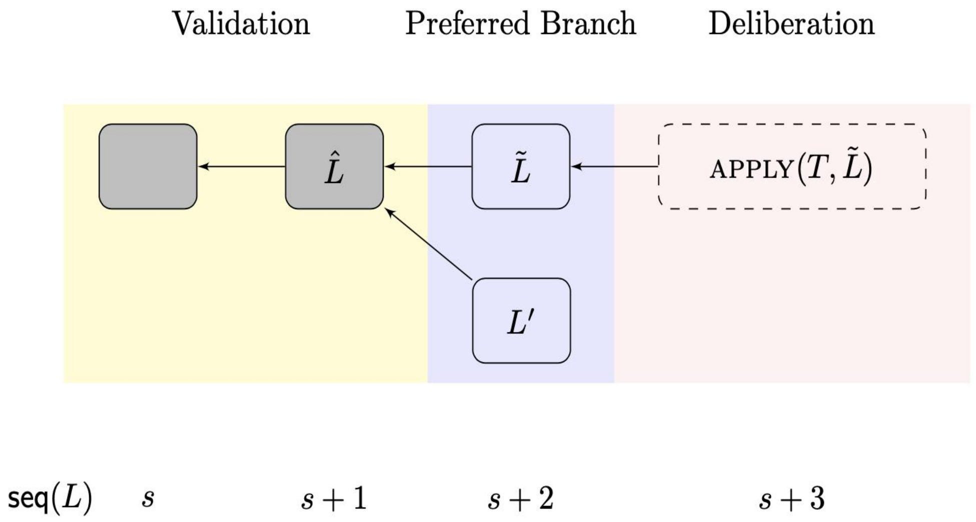


Three Components of XRP LCP

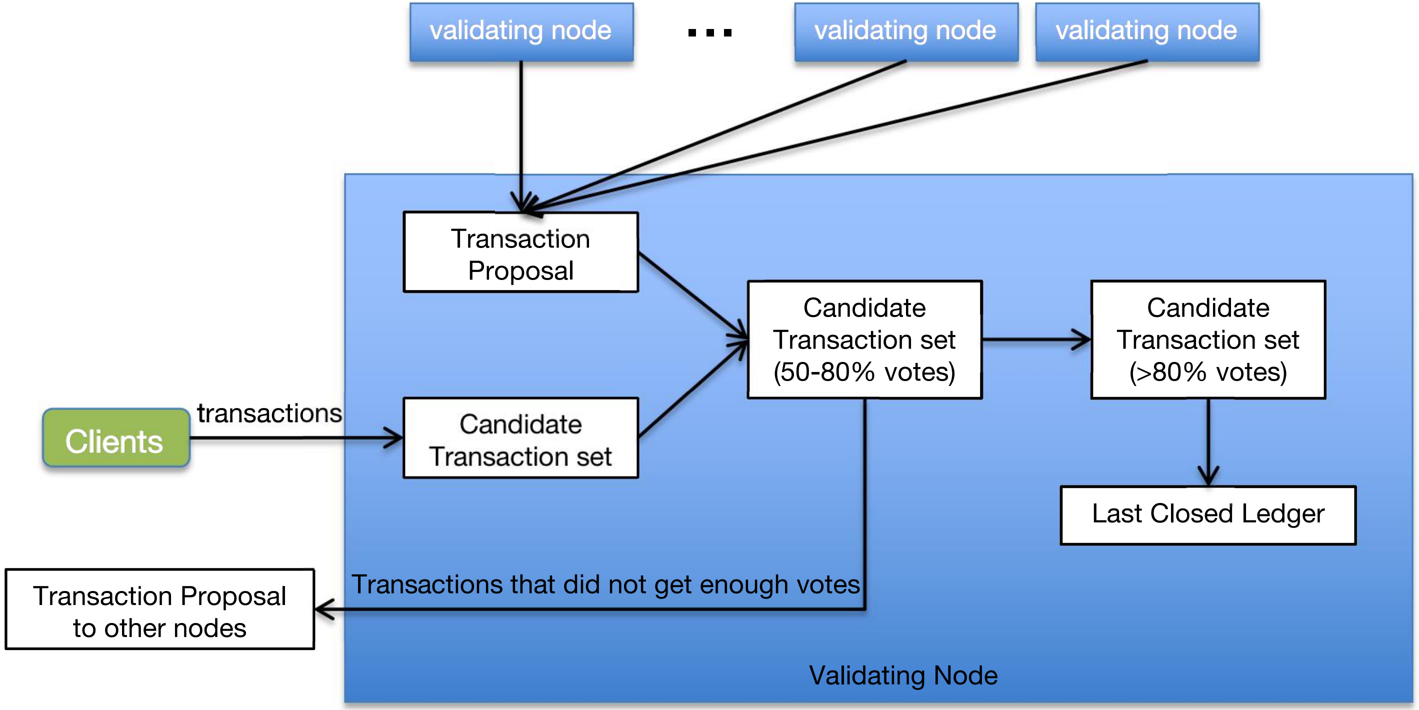
- Deliberation
  - The nodes iteratively propose a transaction set to apply to a prior ledger, based on proposals received from other trusted nodes. When a node believes enough proposals agree, it applies the corresponding transactions to the prior ledger according to the ledger protocol rules. It then issues a validation for the generated ledger.
- validation
  - The node decides whether to fully verify the ledger based on the feedback of the trusted node. Once the quorum of validators is reached, the ledger and its ancestors are considered to have been fully verified, and its status is authoritative and irrevocable.
- Preferred Branch
  - The nodes determine the preferred working branch of ledger history. In times of asynchrony, network difficulty, or Byzantine failure, nodes use the ledger ancestry of trusted validations to resume deliberating on the network’s preferred ledger

Preferred Branch

- **tip support**
  - the number of trusted nodes whose recent validated ledger
- **branch support**
  - the number of trusted nodes whose most recently validated ledger is either L or is descended from L
- **uncommitted support**
  - the number of trusted nodes whose most recent validated ledger is for a ledger with either sequence lower than s or with sequence lower than that of the largest ledger L validation

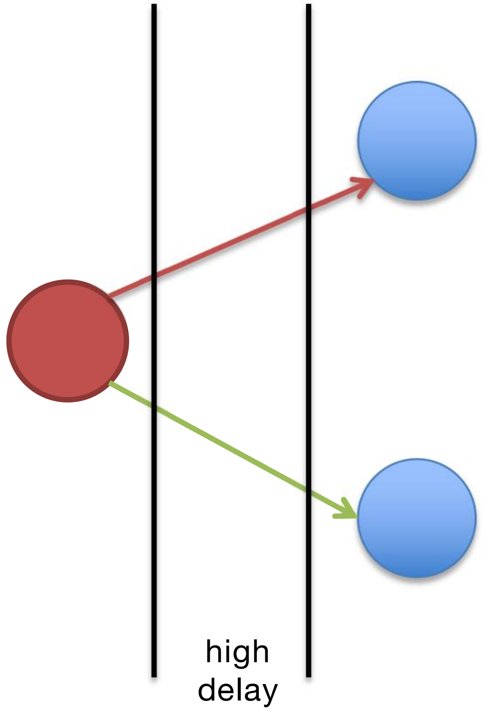


Algorithm Process of XRP LCP

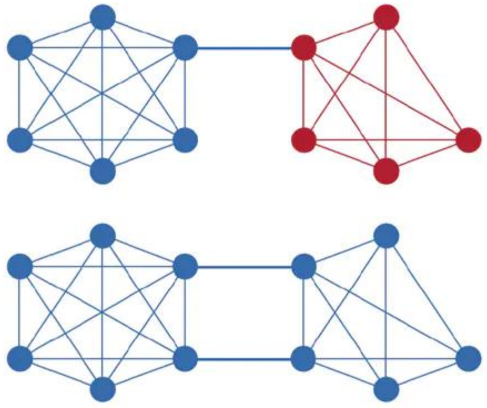

validating node
...
validating node
validating node
Transaction Proposal
Candidate Transaction set (50-80% votes)
Candidate Transaction set (>80% votes)
Clients
transactions
Candidate Transaction set
Last Closed Ledger
Transaction Proposal to other nodes
Transactions that did not get enough votes
Validating Node

Safety of XRP LCP

- Assumption : Byzantine Accountability
  - all nodes – even Byzantine ones – cannot send different messages to different nodes
  - i.e. Byzantine faulty node **cannot** convince two honest nodes that it validated different ledgers
- Under ideally peer-to-peer network——tenable.
- In a fully asynchronous network——untenable.



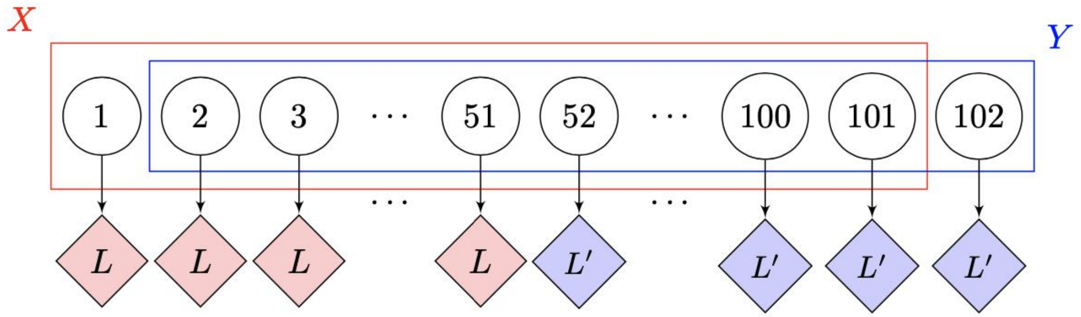
Safety of XRP LCP



- Safety : any completed request will not be changed, it can be seen in future requests
- Fork-safe
  - the minimum overlap requirement was originally believed to be roughly 20% of the UNL
- Once again assuming 80% quorums and 20% faults, the overlap condition can be summarized as requiring roughly > 90% UNL overlaps
- **Cobalt** : An alternative consensus protocol which only needs >60% overlap to match the XRP LCP safety tolerance.

Liveness of XRP LCP

- Liveness
  - can accept and execute requests from non-Byzantine clients, and will not be affected by any factors that cause the requests of non-Byzantine clients to be unable to be executed
- stuck network with 99% UNL overlap and no byzantine faults



Liveness of XRP LCP

- XRP trust model has all nodes listening to either a single UNL consisting of 5 nodes, or a UNL consisting of those 5 nodes plus one extra node(typically the extra node is one- self; nodes that listen to these extended UNLs are thus called ”leaves”)

- In the restricted case of a single expanding UNL with leaves, we can always make forward progress during periods when no nodes are faulty and network messages are delivered with bounded delay. In the more general case with even minor disagreement of participants, we cannot guarantee that the network makes forward progress.