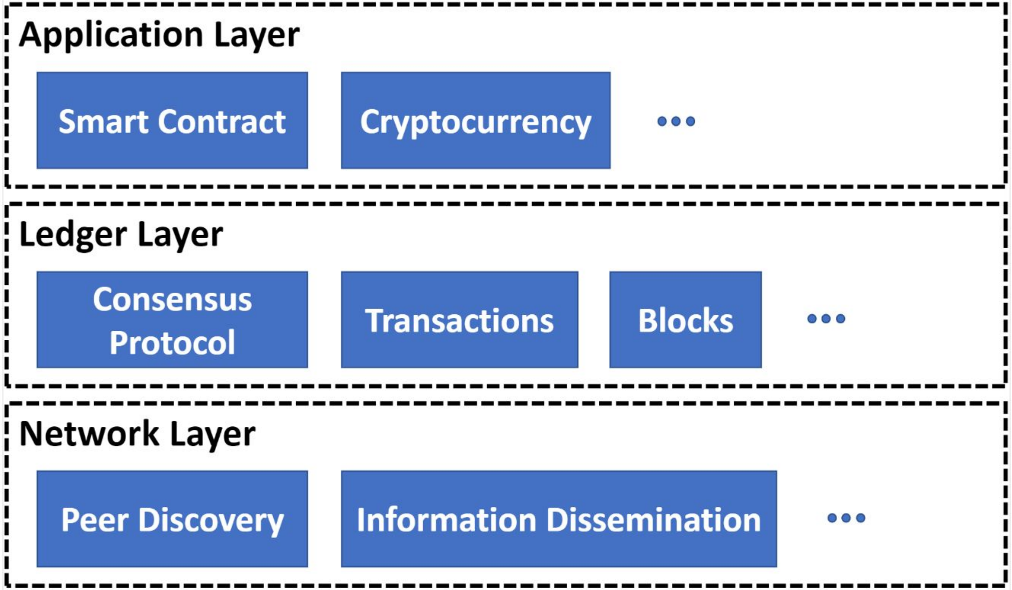
FRing: A P2P Overlay Network for Fast and Robust Blockchain Systems

Haoran Qiu, Tao Ji

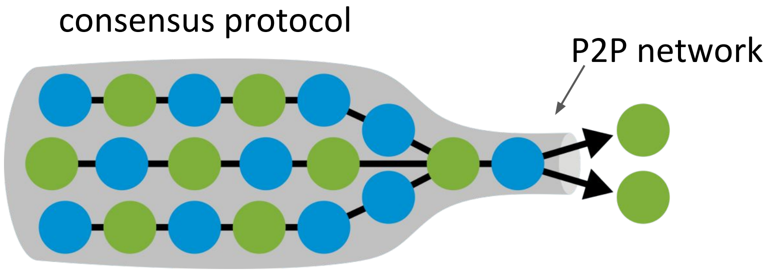
HKU System Group
Department of Computer Science

Blockchain Systems

- **Layered** structure
  - Application layer
  - Consensus layer
  - P2P overlay network layer
  - OS Network subsystem



Research Question



- Bitcoin is slow (up to 7 Tx/s)
- Ethereum is not much better (10~30 Tx/s)
- However, many blockchain systems claims to achieve **2K~10K** Tx/s:
  - EOS, HLF, NEO, Conflux, Omniledger, etc.
- Current network layer of blockchain systems work well for Bitcoin and ETH.
- However, higher transaction rate -> higher broadcast frequency
  -> larger bandwidth and shorter convergence time required
- Unfortunately, P2P network have become the **bottleneck** of higher transaction rates

Problem of Current P2P Overlay Networks

- **Network topology** - formed during peer discovery
  - Random graph, e.g. Bitcoin
  - DHT-based graph (essentially random), e.g. Ethereum

- **Long convergence time** for broadcasts
  - broadcast topology formation does not consider geographical proximity
  - high-latency paths are incurred
  - worst case: frequent jumping between two components that are far away from each other

$I = \{0, 1, \ldots, 2^b - 1\}$

$i \in I$

$f : I \to I^k$

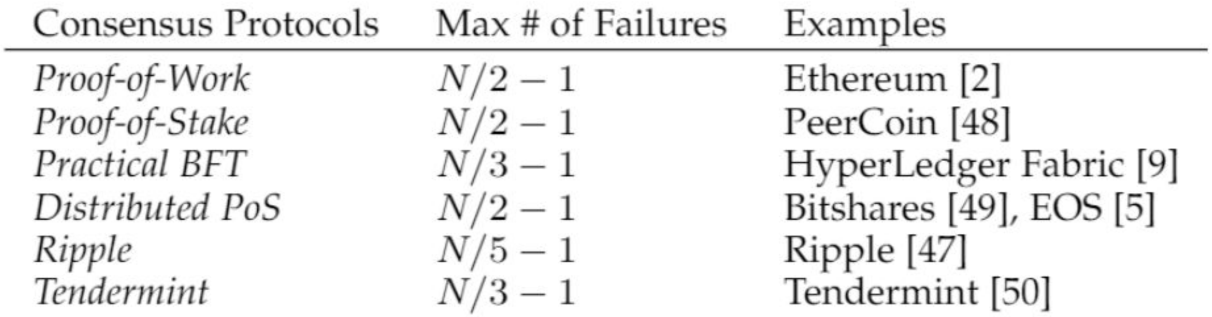
Problem of Current P2P Overlay Networks

- **Broadcast**
  - Dominant: Gossip-based broadcast
    - Push / Pull versions
    - Other variants: TTL, UMID, central server, etc.
  - Tree-based broadcast
    - ByzCoin

- Gossip generates excessive **redundant messages** for extreme robustness (90%)
  - traffic congestion (msg accumulation)
  - exacerbated when network bandwidth is low or broadcast frequency is high

Design Insights #1

- Gossip is **overly robust** for blockchain systems
  - all state-of-the-art blockchain systems can only tolerate **20%-50%** failure
  - Gossip can tolerate up to **90%** failure



Consensus Protocols	Max # of Failures	Examples
Proof-of-Work	$N/2 - 1$	Ethereum [2]
Proof-of-Stake	$N/2 - 1$	PeerCoin [48]
Practical BFT	$N/3 - 1$	HyperLedger Fabric [9]
Distributed PoS	$N/2 - 1$	Bitshares [49], EOS [5]
Ripple	$N/5 - 1$	Ripple [47]
Tendermint	$N/3 - 1$	Tendermint [50]

Design Insights #2

- Taking **geographical locality** into consideration reduces convergence time
    - incur low latency paths
    - avoid unnecessarily high latency paths
- High level idea:
    - Group nodes that are geographically close to each other together
    - Representatives are used for communication between two groups

Design Insights #2

- Problem:
  - possible eclipse attack on victims in a group
  - risk of topology inference by traffic pattern analysis
- Mitigation:
  - Intel SGX
  - Pattern obfuscation

Summary on Existing P2P Networks

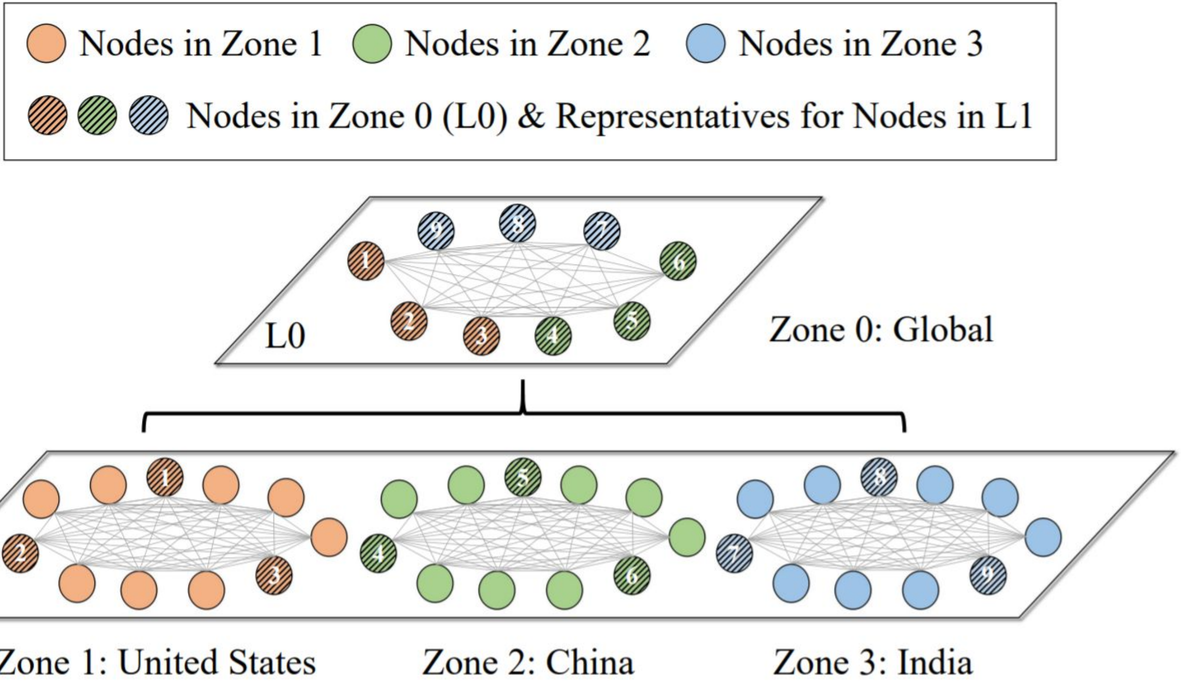
	Message Redundancy	Convergence time	Robustness
Gossip-based	O(NlogN)	Slow, non geo-based, probabilistic	Extreme robust, tolerate up to 90%
Tree-based	O(N), optimal	Medium, non geo-based, deterministic	Low, tolerate only leaf node failure
FRing	O(N), optimal	Fast, geo-based, deterministic	Sufficient for all blockchain systems

FRing’s Features

- Fast convergence
  - low-latency paths have higher priority than the high-latency ones
  - accumulation of old messages is reduced effectively
- Low message redundancy
  - O(N)
- Sufficient robustness
  - a broadcast operation can tolerate at least the same portion of node failure as consensus protocols in blockchain systems

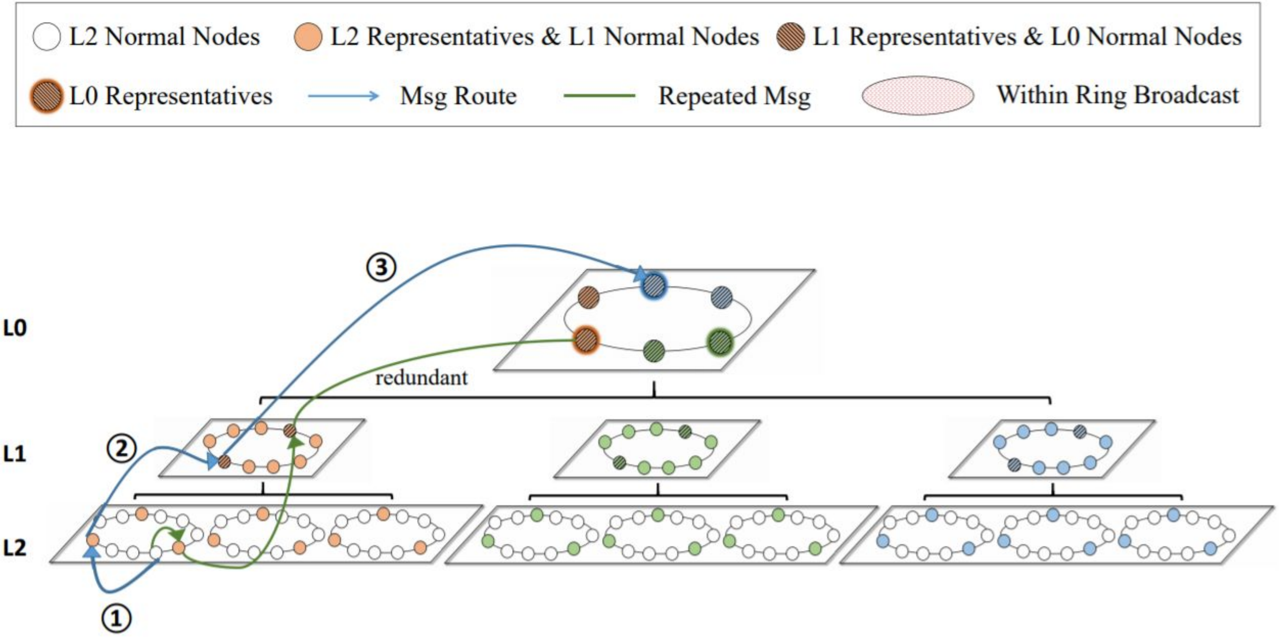
FRing's Topology

- Fractal rings
- Hierarchical structure
- Recursive
- Geography-based



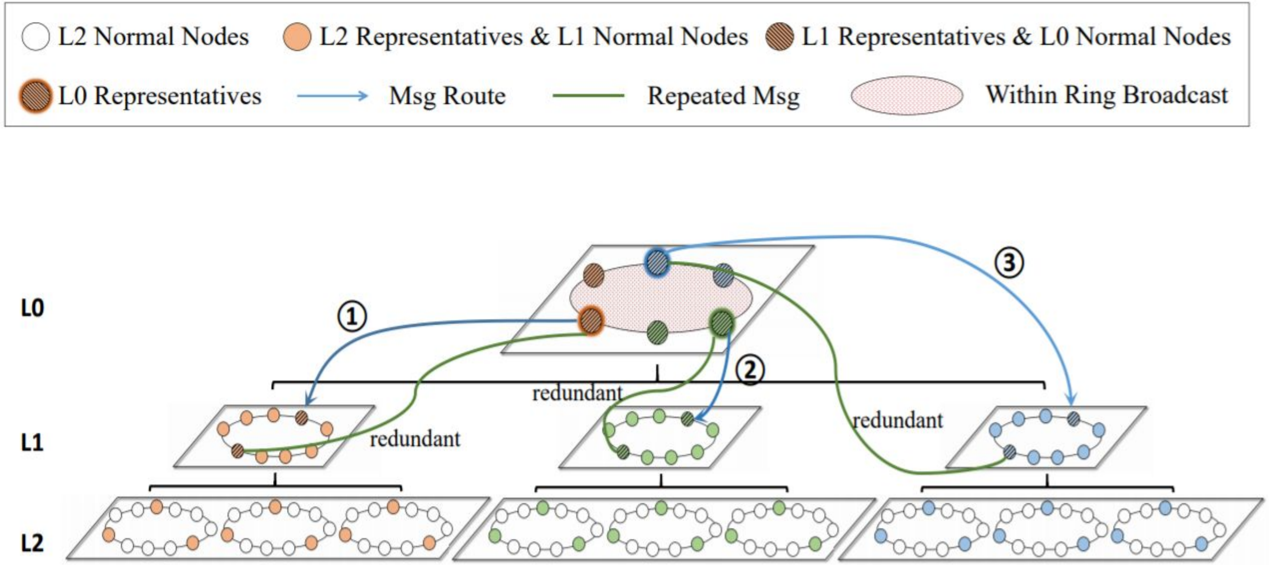
FRing’s Broadcast Mechanism

- Broadcast
  - **upwards**
  - downwards
  - within-ring



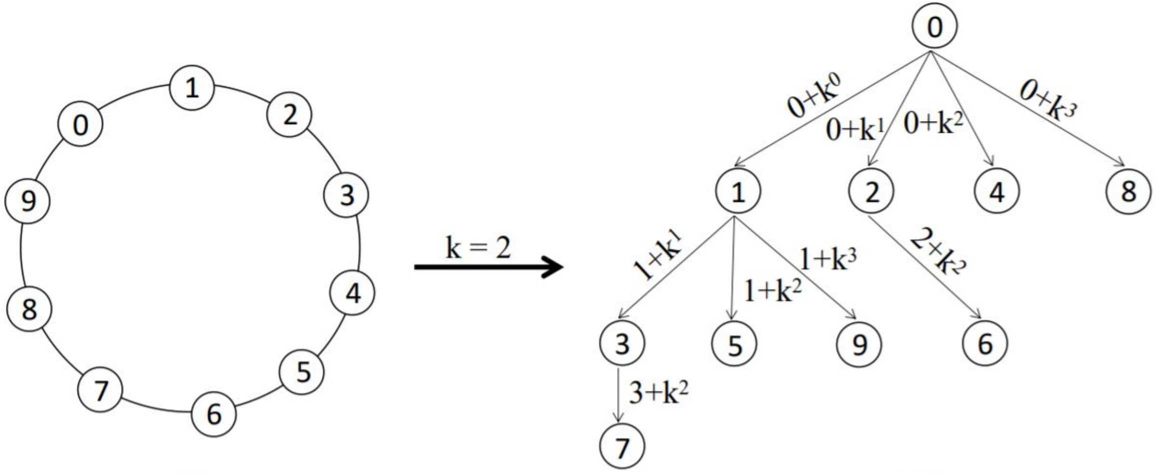
FRing’s Broadcast Mechanism

- Broadcast
    - upwards
    - **downwards**
    - within-ring



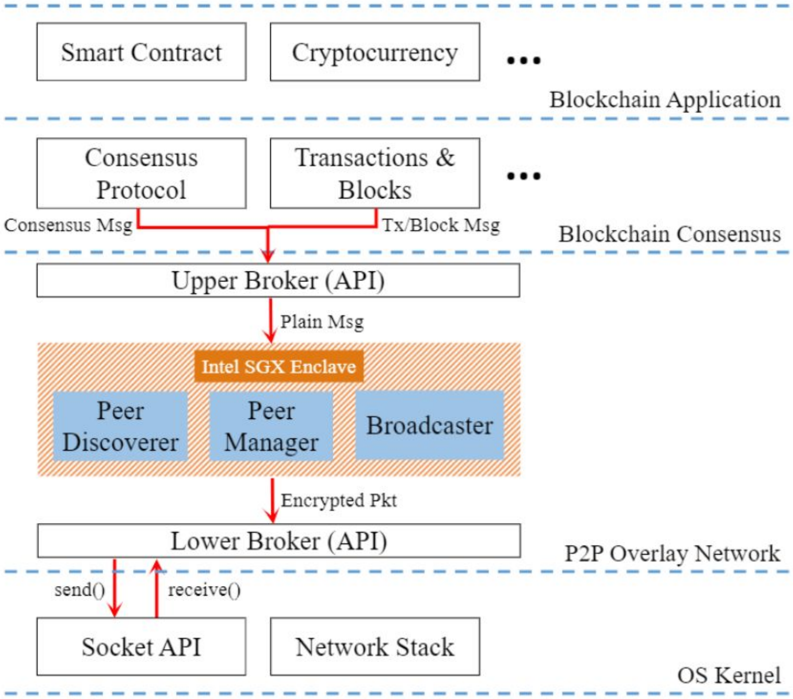
FRing’s Broadcast Mechanism

- Broadcast
    - upwards
    - downwards
    - **within-ring**, i.e. k-ary distributed spanning tree



(a) (b)

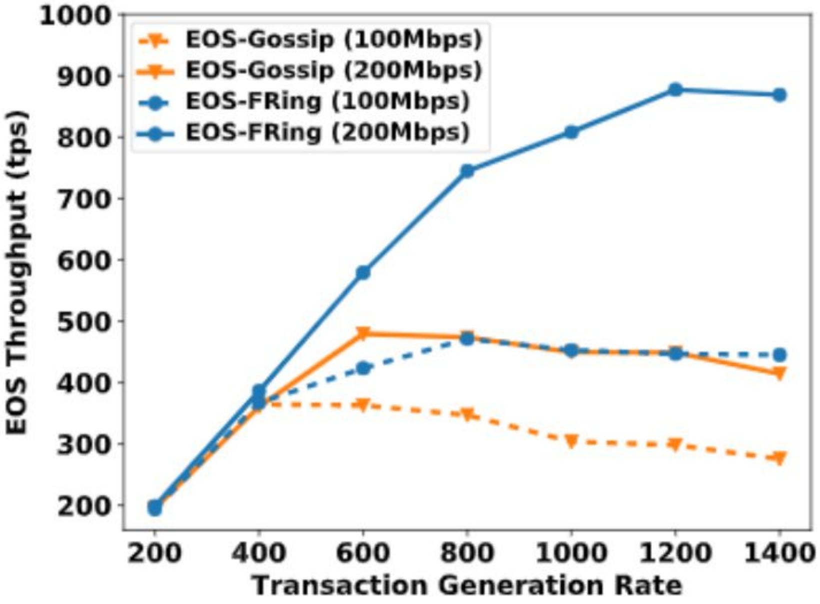
Architecture of FRing

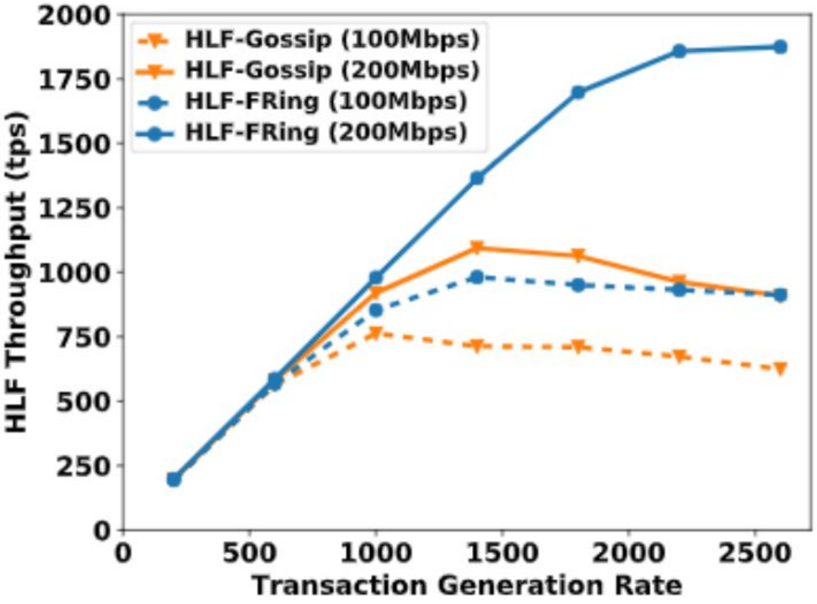


Evaluation

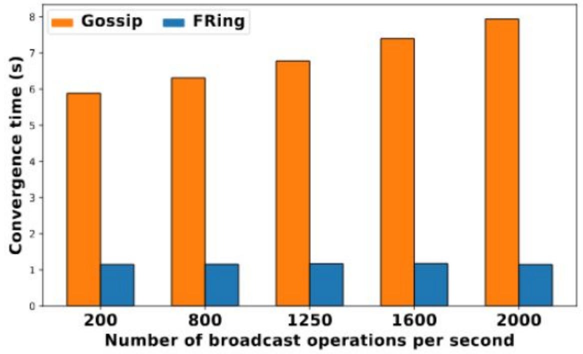
- Evaluation questions:
  - How effective can FRing improve the **end-to-end** performance?
  - How effective can FRing reduce the **message** complexity and convergence **time** for broadcast? Is FRing **scalable**?
  - Can FRing provides **sufficient fault-tolerance** for blockchain systems?
  - Can FRing prevent representative nodes from detection?
- Evaluation setting:
  - up to 8000 nodes with Docker in AWS
  - 30 c4.4xlarge VMs with 16 cores and 30 GB memory in the same region
  - simulate RRT latency between cities, states, countries (7 layers)

End-to-end Throughput

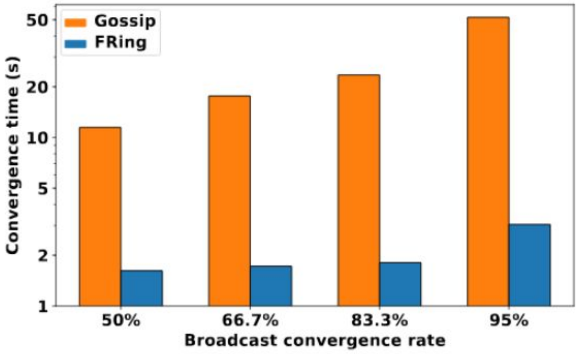




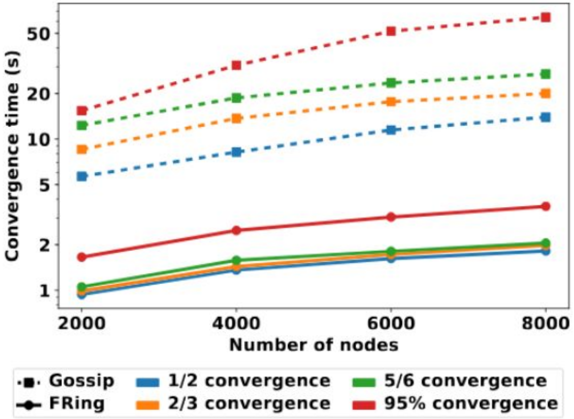
Convergence Time



(a) Convergence time comparison with respect to the broadcast rate for 6K nodes and 2/3 convergence rate.

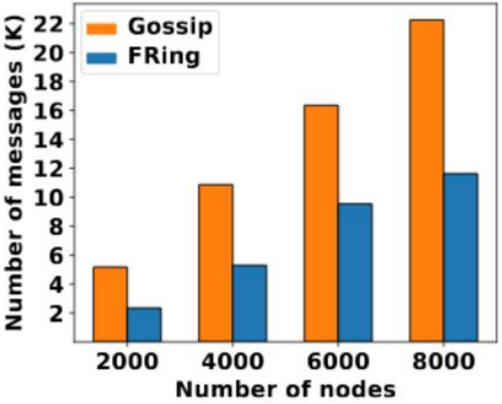


(b) Convergence time comparison with respect to the convergence rate for 6K nodes and 200 Tps Tx generation rate.

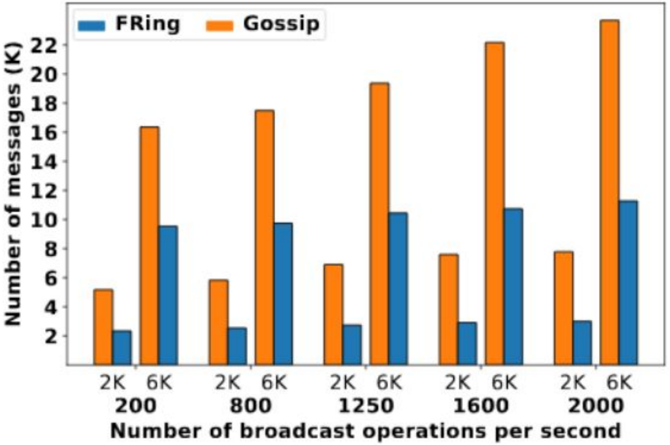


(c) Scalability analysis on number of nodes with 200 tps Tx generation rate.

Message Complexity

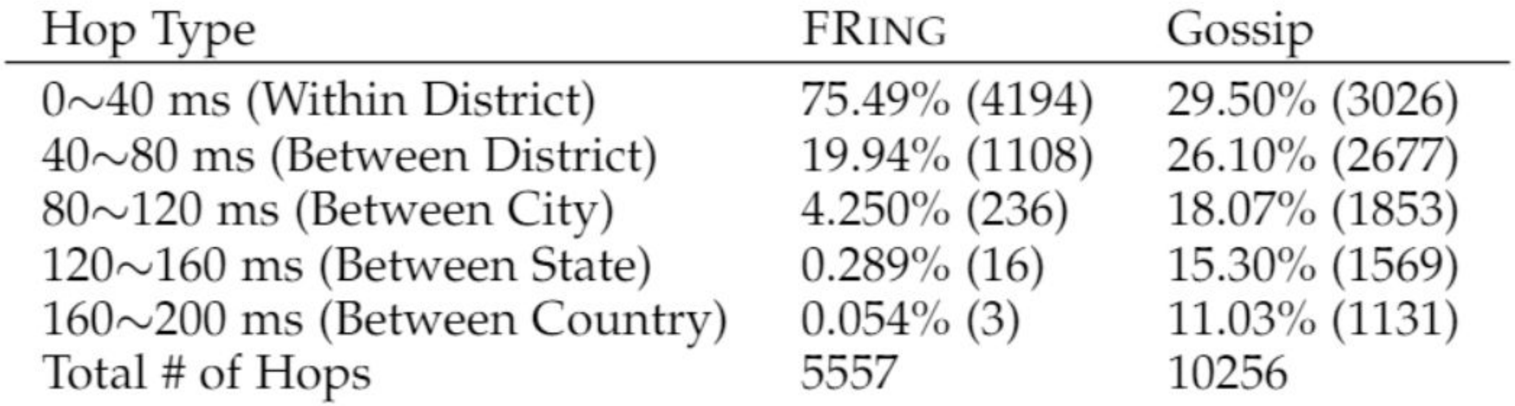


(a) Message complexity with respect to the number of nodes under 200 tps transaction rate.



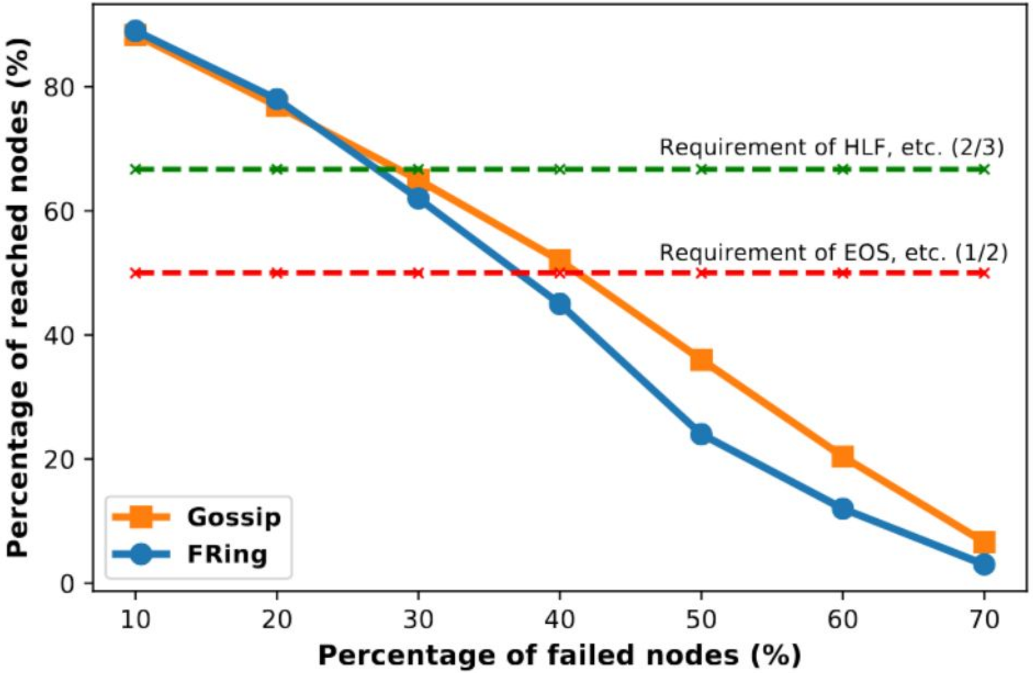
(b) Message complexity with respect to broadcast operation rate from 200 tps to 2000 tps.

Convergence Time - hop analysis

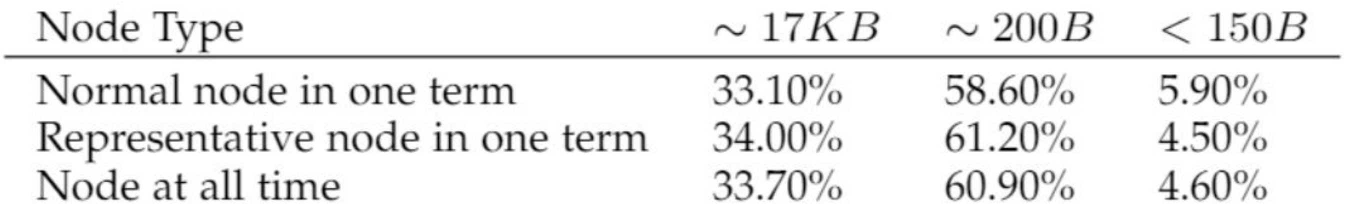


Hop Type	FRING	Gossip
0~40 ms (Within District)	75.49% (4194)	29.50% (3026)
40~80 ms (Between District)	19.94% (1108)	26.10% (2677)
80~120 ms (Between City)	4.250% (236)	18.07% (1853)
120~160 ms (Between State)	0.289% (16)	15.30% (1569)
160~200 ms (Between Country)	0.054% (3)	11.03% (1131)
Total # of Hops	5557	10256

Fault-tolerance for Node Failures

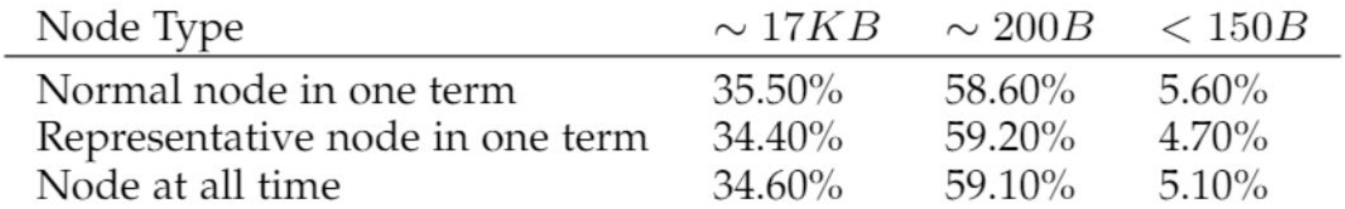


Traffic Analysis



Node Type	$\sim 17KB$	$\sim 200B$	$< 150B$
Normal node in one term	33.10%	58.60%	5.90%
Representative node in one term	34.00%	61.20%	4.50%
Node at all time	33.70%	60.90%	4.60%

TABLE 4: Send-Packet Analysis of Node in FRING



Node Type	$\sim 17KB$	$\sim 200B$	$< 150B$
Normal node in one term	35.50%	58.60%	5.60%
Representative node in one term	34.40%	59.20%	4.70%
Node at all time	34.60%	59.10%	5.10%

TABLE 5: Receive-Packet Analysis of Node in FRING

Conclusion

- FRing is the first geography-based P2P overlay network that achieves fast and robust broadcast for blockchain systems.
- By trading off excessive robustness and considering geographical locality, FRing improves the throughput of blockchain systems by increasing broadcast message efficiency and convergence time.
- Evaluation and analysis show that FRing is efficient, sufficiently robust, and secure.
- FRing has the **potential** to facilitate the development of blockchain consensus protocols with even higher transaction rates.

Discussion/Future directions

- Does FRing has the potential to facilitate blockchains with **sharding**? Attacks?
- FRing improves the efficiency of blockchains, what about **security/anonymity**?
- **Alternative design/solution** to solve the over-robust problem of Gossip?
- Is a general network the optimal fit for **heterogeneous** blockchains? or a network layer should also be heterogeneous?

Thank you!