Enhancing Scalability and Liquidation in QoS Lightning Networks

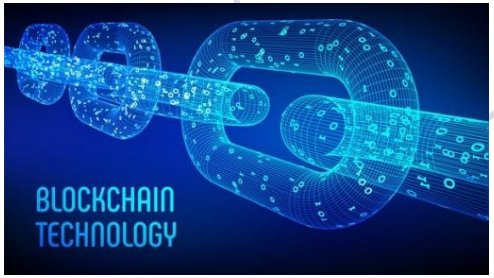
Jie Wu

Department of Computer and Information Sciences

Temple University, USA

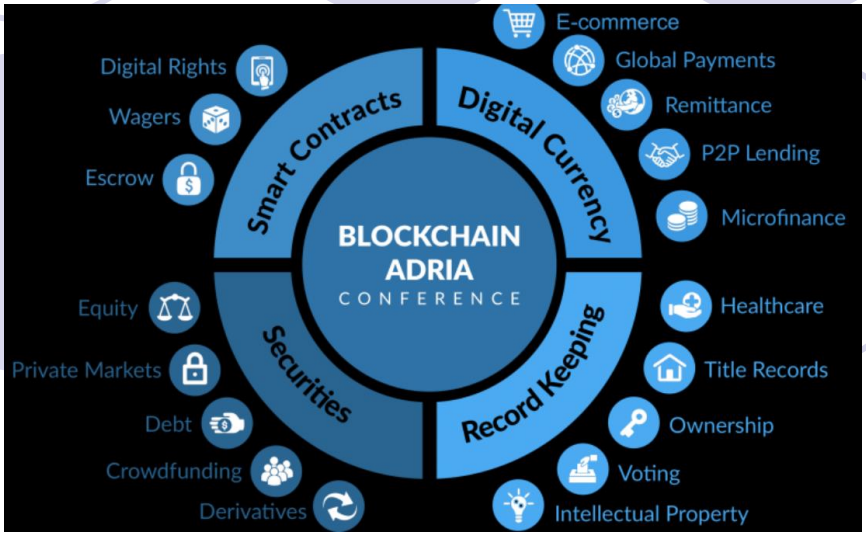
Outline

1. Blockchain and Lightning Networks
2. QoS: Scalability and Liquidation
3. Supernode-based Clustering
4. Pooling and Pruning
5. Performance Evaluation
6. Future Work



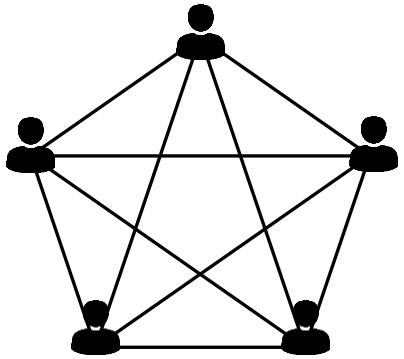
1. Blockchain and Lightning Networks

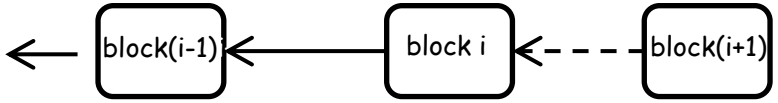
- Blockchain
  - A system of maintaining transactions in a P2P network
  - Distributed ledger
- Bitcoin
  - Bartering
  - Metallic money
  - Paper money
  - Cryptocurrencies



Blockchain Basics

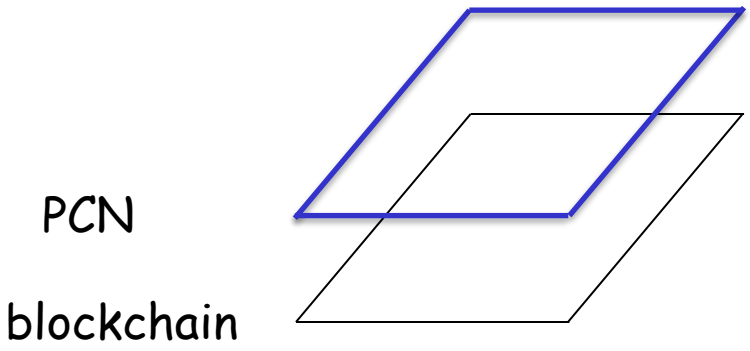
- Components
  - Transaction/block, incentive, and consensus
- PoW-based blockchain mining
  - Mining a block: puzzle solving (Nakamoto protocol)
  - PoW: Prob. of solving a puzzle (computing rate)
  - Individual chaining of blocks





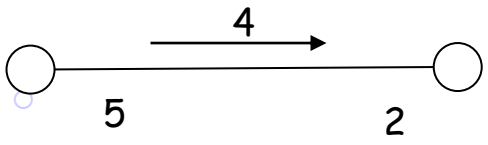
Blockchain Scalability

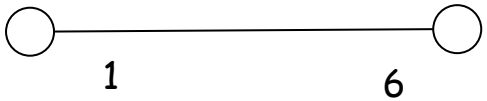
- Scalability problem
  - 10 minutes per block (1 MB) or ≤ 7 transactions per second
- Solutions
  - On-chain: block-size, sharding, other consensuses (PoS/PoC)
  - Off-chain: SegWit, side chain, and tree chain
- Payment channel network (PCN): layer 2



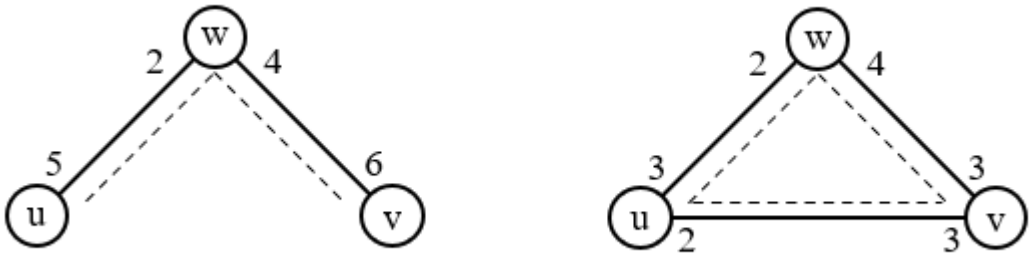
Lightning Networks (LNs)

- Micropayment channels
  - Quick transactions between trusted neighbors
  - Avoiding block confirmation via off-chain payment
- Fund allocations
  - Allocation of node funds to channels
- Bidirectional transactions
  - Fund balance in two directions: *channel capacity*
  - Channel balance of two sides are private





Payment Path

- Indirect fund transfer
  - Between two untrusted neighbors
- Payment path
  - A sequence of non-repeated trusted neighbors
- Types of paths
  - Single-path and multi-path
  - e.g., transfer $4 from u to v
  - 

2. QoS: Scalability and Liquidation

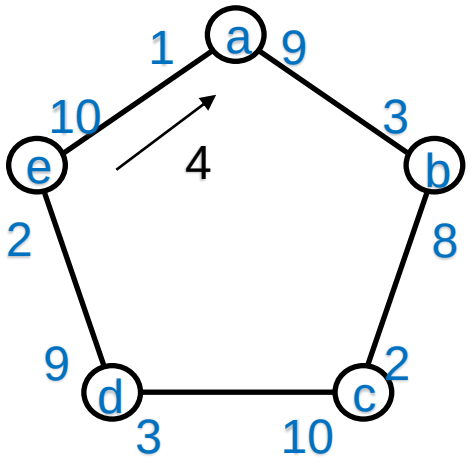
- Scalability for path searching
  - Searching process with global information
    - Route validation (for channel balance check)

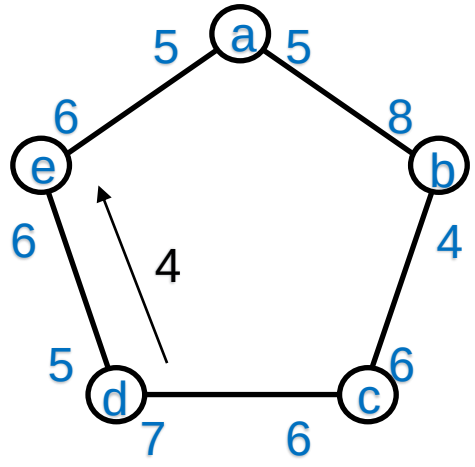
- Liquidation for fund transfer
  - Success ratio for transactions
    - Alleviated with multi-path, but more involved

LN is dynamic: A change in topology or capacity is broadcasted

Existing Solutions

- Current scalability solution
  - Flare: reducing time to find a payment route
  - Challenges: large search space
- Current liquidation solution
  - Revive: rebalancing cycles
  - Challenges: fixed channel capacity



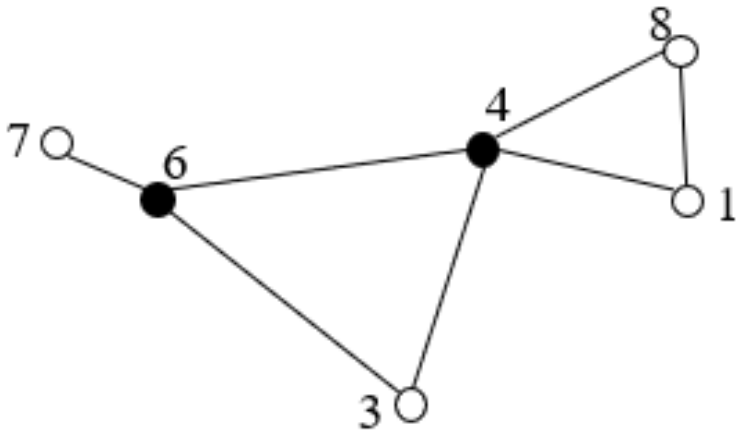


3. Supernode-based Clustering

- A special clustering approach
  - Addressing both scalability and liquidation
  - Graph $G = (E, V)$ partitioned into clusters locally (why?)
- Supernodes $S \subset V$
  - Each cluster is headed by a supernode
  - Being locally self-connected reduces update cost

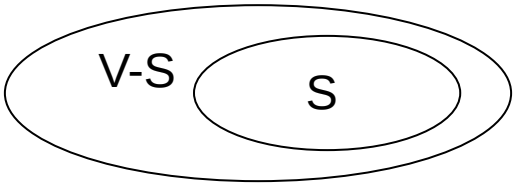
Clustering

- Basic algorithm
  - Node v is clusterhead when v has two unconnected neighbors
- Property
  - Clusterheads form a locally self-connected dominating set (DS)
  - Low-efficiency: too many clusterheads (black nodes)



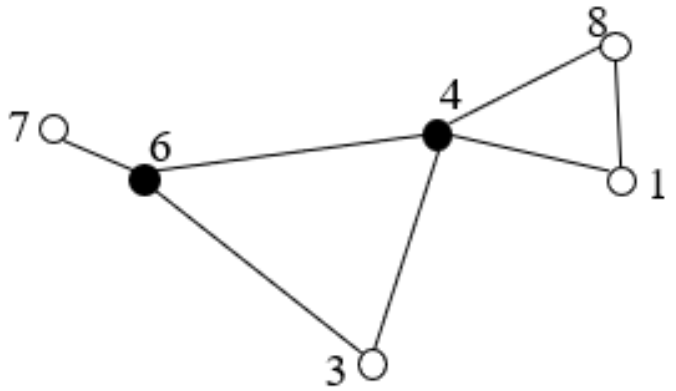
4. Pooling and Pruning

- Supernode selection (S)
  - Induced subgraph G[S] is connected and V-S $\subseteq$ N(S)
- Fund pooling by supernodes



  - Pooling funds in all clusters
  - Redistributing funds to external channels in G[S]
- Routing
  - Searching in a reduced space in G[S], rather than G

Design Details



- Each node
  - Knows its k(=2)-hop info.
- Escrow Account
  - Each node has one escrow account for its supernode neighbor
- Fund allocations of supernodes
  - Use escrows to allocate more funds to external channels
- Implementation
  - Local status calculation, then status/link-state broadcasting

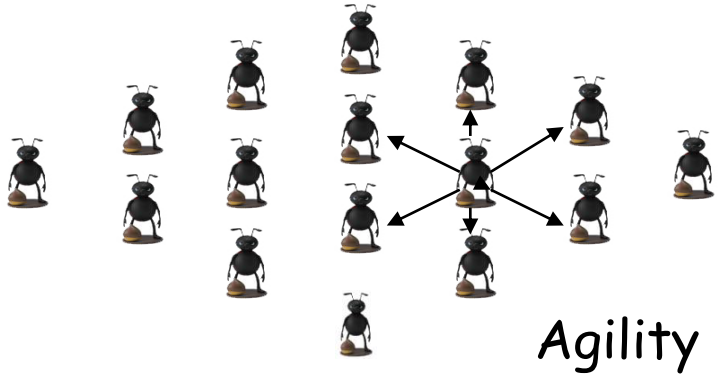
Self-Organizing Local Solutions

Local decisions/fixes

- P2P and simple interaction (local w/o seq. propagation)

Global functionality

- E.g., connectivity



Agility

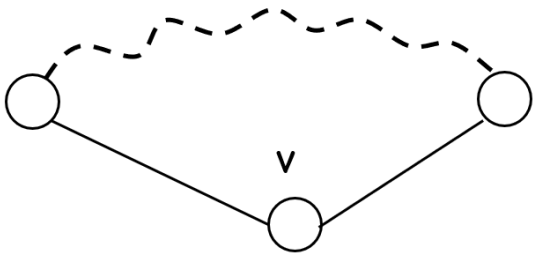
Principles

- P_1: Local actions w/ global properties (scalability)
- P_2: Minimization of maintained state (usability)
- P_3: Adaptive to changes (self-healing)
- P_4: Implicit coordination (efficiency)

Node Pooling

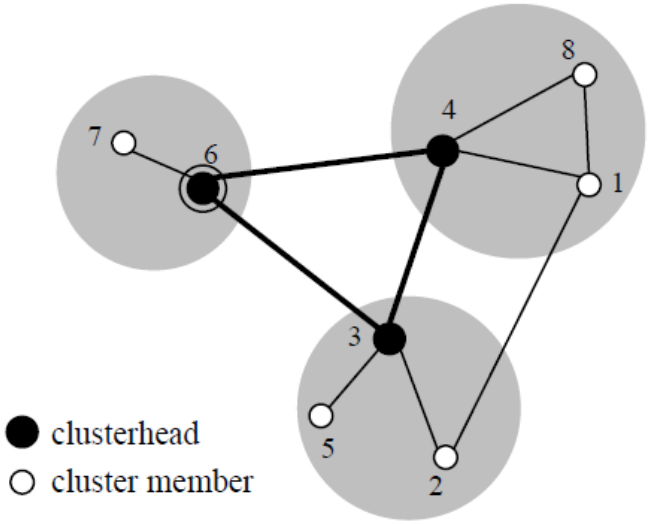
- Supernode selection (Wu and Dai 2004)
  - All nodes are initially supernodes
  - A supernode v becomes a non-supernode if any two neighbors of v are connected by a path (under k-hop view, k=2) such that for each intermediate node u in the path, Pri(u) > Pri(v)

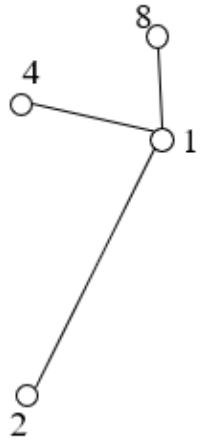
- Time complexity: $O(\Delta^2)$, where Δ is max node degree

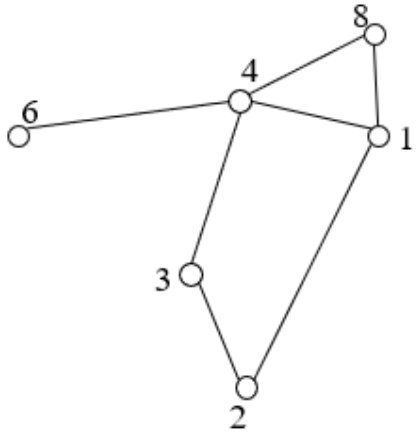


Supernode Selection

- Node v
  - 2-hop local view
  - A distinct priority Pri(v)
- Supernode set
  - S = {3, 4, 6}
- Node 1's view
  - 1-hop view
  - 2-hop view







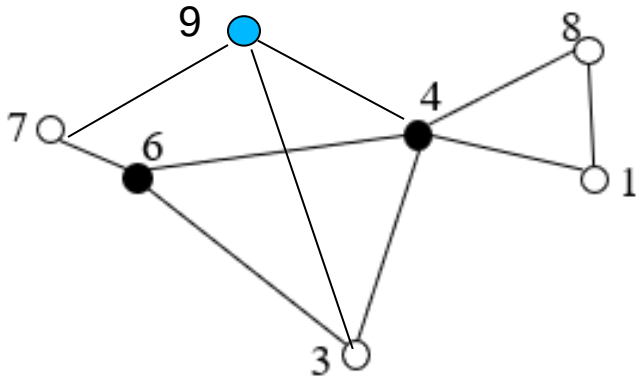
Network Dynamics

- Nodes joining/leaving (channels up/down)
  - Local update (2-hop): no propagation
  - Supernode stability: graceful evolving clustering

Theorem: When a node is added to/deleted from an LN, it will only affect the status of k-hop neighborhood of the node.

e.g., deleting 3 changes no node & adding 9 changes 6 to a regular node

- 4

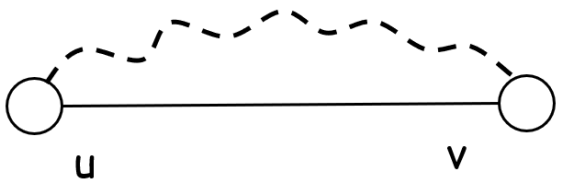


Link Pruning

- Link Priority
  - Reserve lexicographical order for link uv

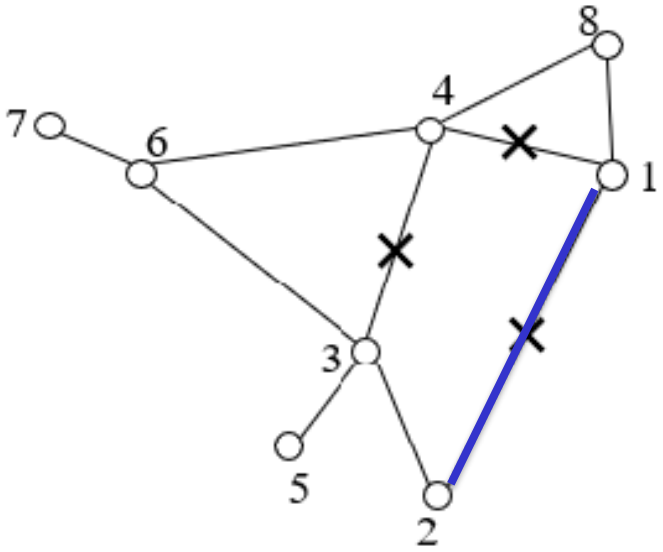
    Pri(max{Pri(u), Pri(v)}, min{Pri(u), Pri(v)}), e.g., Pri(3, 2) > Pri(3, 1)

- Link Pruning (Wu and Jiang 2020)
  - Link uv can be removed if there is a replacement path (under k-hop view, k=2) connecting u and v: all intermediate links have higher priorities than uv.



Neighbor Set Reduction

- Asynchronous link pruning
- Still 2-hop views from two end nodes
- Replacement paths (avoiding circular replacements)



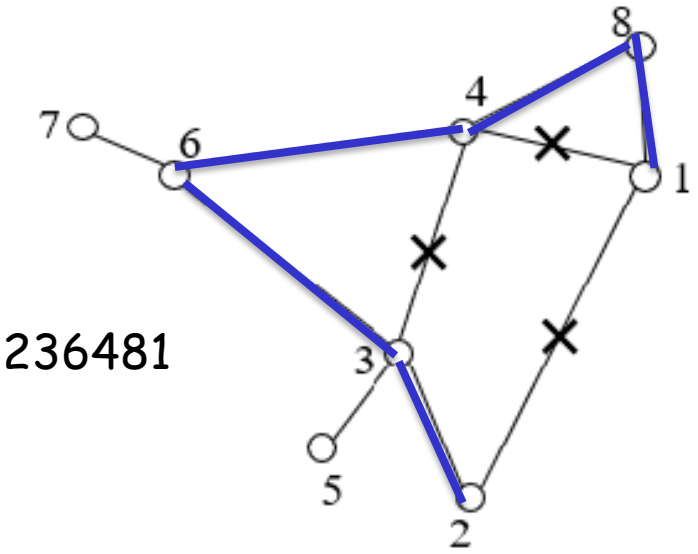
Replace for 12:

2341
23481
23641
236481

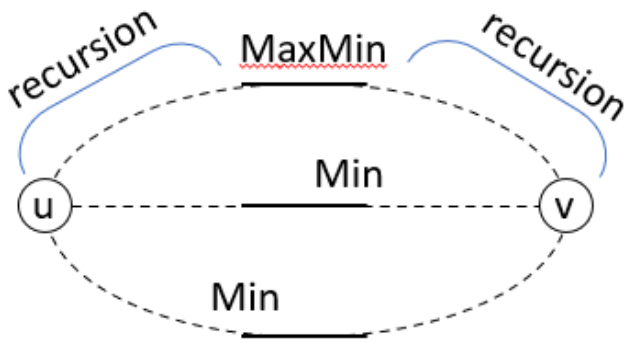
Irreplaceable Replacement Path

- Irreplaceable replacement path
  - Min link: of a replacement path
  - MaxMin link: max of min links of all replacement paths

Theorem: Given a connected graph, the resultant graph after link pruning will remain connected.

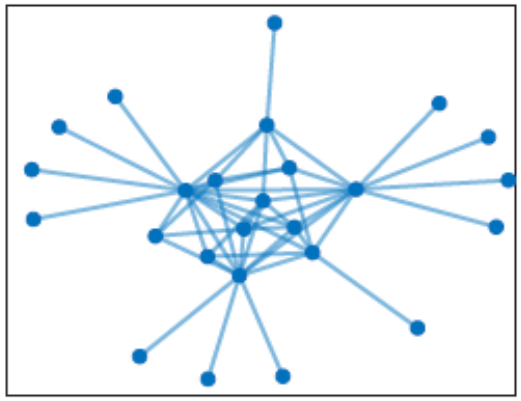


236481

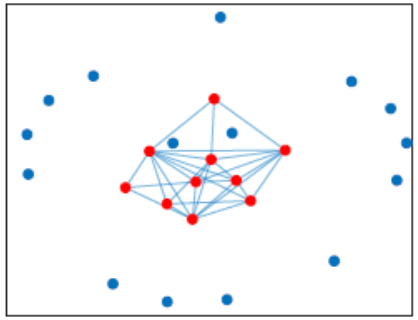


A 25-node example

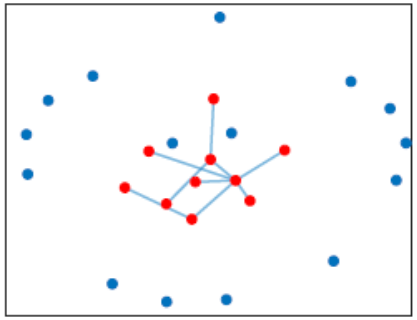
- Pooling or pruning only
- A combination of pooling and pruning



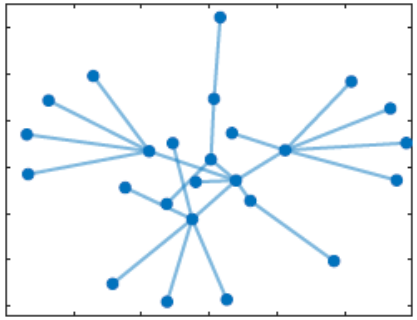
(a) The original LN topology.



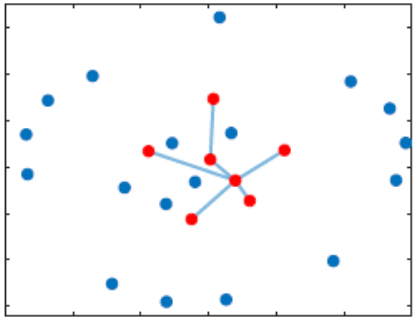
(b) Pooling only.



(c) Pooling then pruning.



(d) Pruning only.



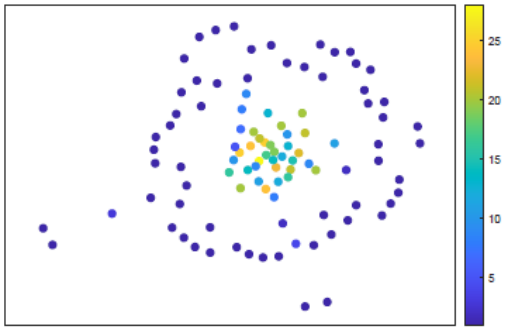
(e) Pruning then pooling.

5. Performance Evaluation

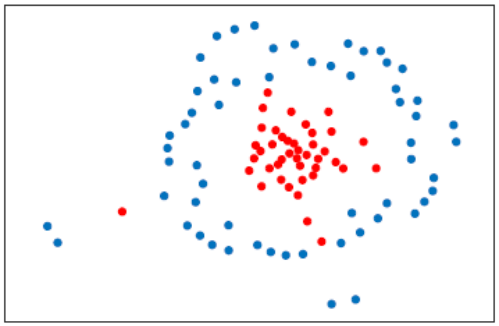
- Channel capacity
  - Three intervals
- Channel balance
  - Perfectly balanced
  - Randomly balanced
- Transaction amount
  - Homogeneous
  - Heterogeneous
    (micro, small, med., large)
- Node/link reduction
  - Pooling/pruning efficiency
- Success ratio (SR)
  - Single transaction (ST)
  - Transaction flow (TF)
- Path length (PL)
  - Routing fees (not include)
- Node degree (ND)

Topologies

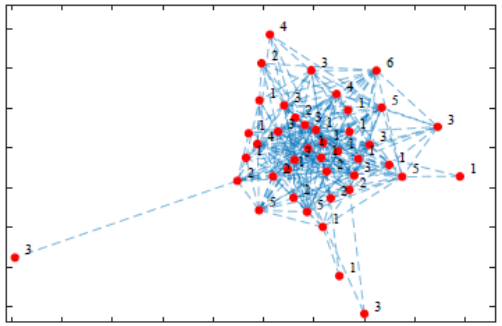
- Custom network (CN)
  - Power law distribution for LNs
  - 100 nodes and 340 links
  - Reduction to 42 supernodes
- ISP and Watts-Strogatz (WS)
  - ISP: power law and WS: small world



(a) The custom network.



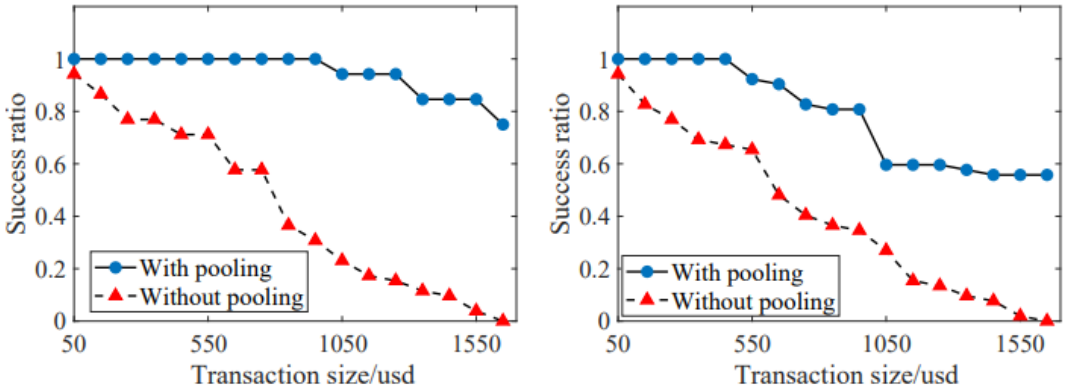
(b) Pooling: supernodes marked red.



(c) Supernodes: numbers for pool sizes.

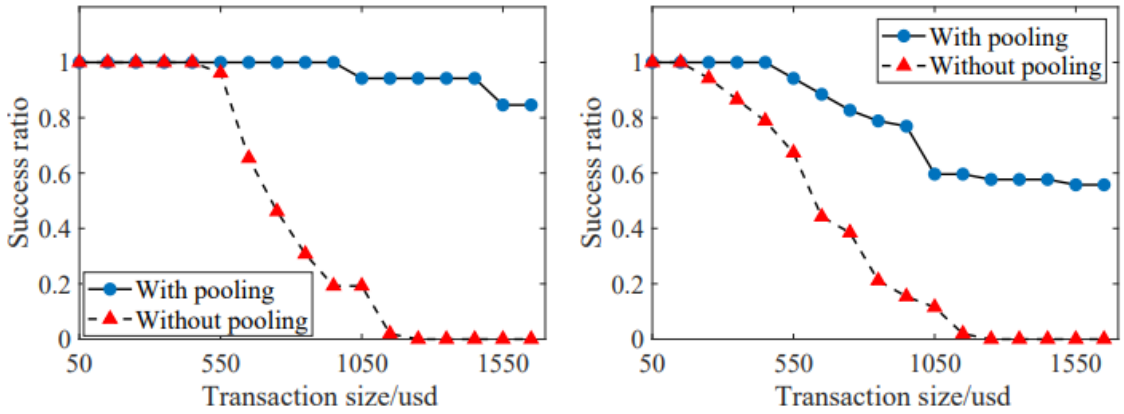
Pooling only on CN: homogenous

Randomly balanced



(a) Single transaction success ratio. (b) Transaction flow success ratio.

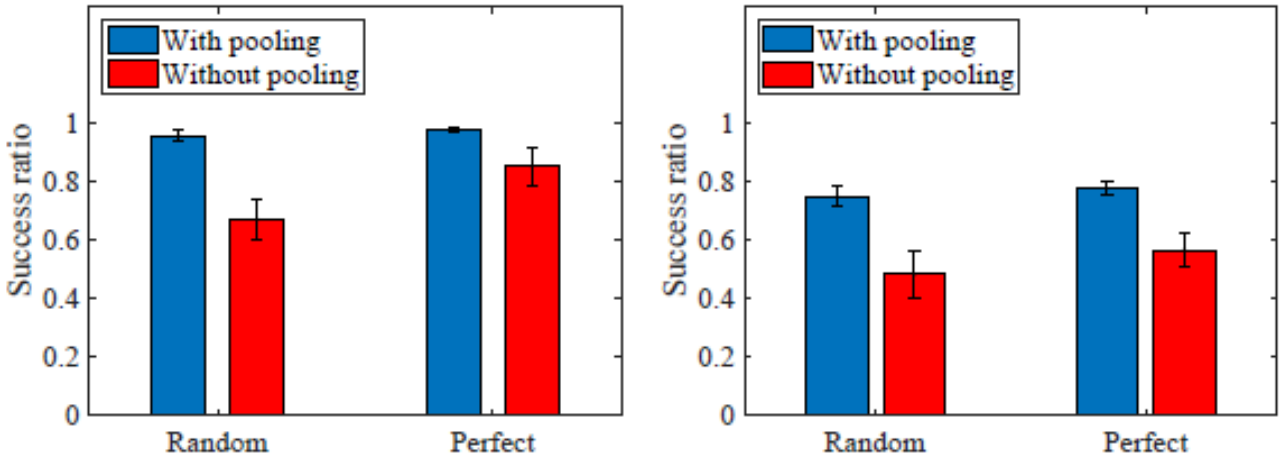
Perfectly balanced



(a) Single transaction success ratio. (b) Transaction flow success ratio.

Channel capacity: [1000, 1500) 50%, [1500, 2000) 35%, [2000, 2500) 15%

Pooling only on CN: heterogenous



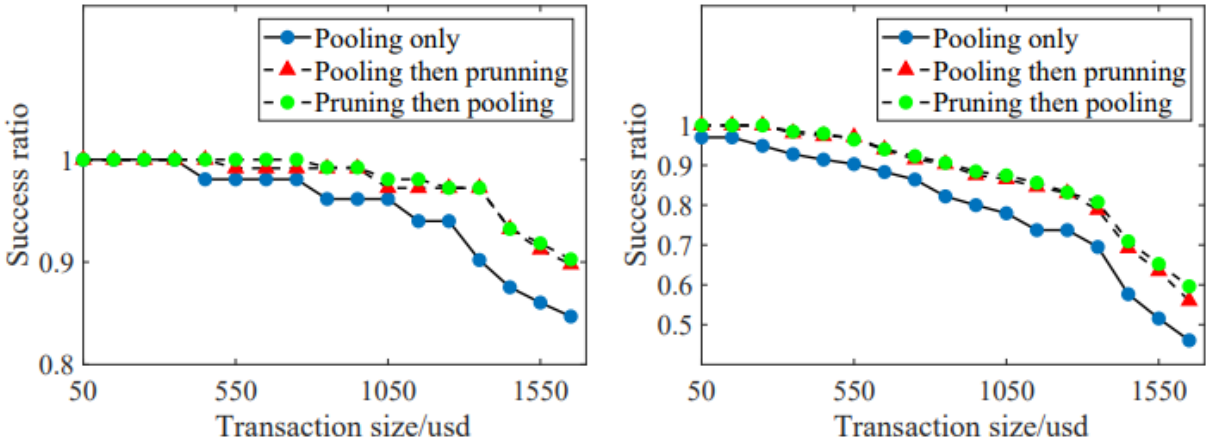
(a) Single transaction success ratio. (b) Transaction flow success ratio.

Improvement is more significant in transaction flow

Transactions: micro (0, 200] 40%, small (200, 800] 30%
med. (800, 1000] 20%, large (1000, 1600] 10%

Pooling + Pruning on CN

Homogenous



Heterogenous

$(\|V\|, \|E\|)$	Operation	STSR	TFSR	PL	ND
$(100, 340)$	W/O pooling	0.45	0.48	4.89	6.80
$(42, 255)$	W/ pooling	0.75	0.77	6.35	12.14
$(42, 213)$	Pooling, pruning	0.88	0.83	7.01	10.14
$(43, 226)$	Pruning, pooling	0.87	0.86	7.12	10.51

ISP and WS

Effectiveness of pooling/pruning on success ratio (SR)

Tradeoff: SR vs. path length (PL)

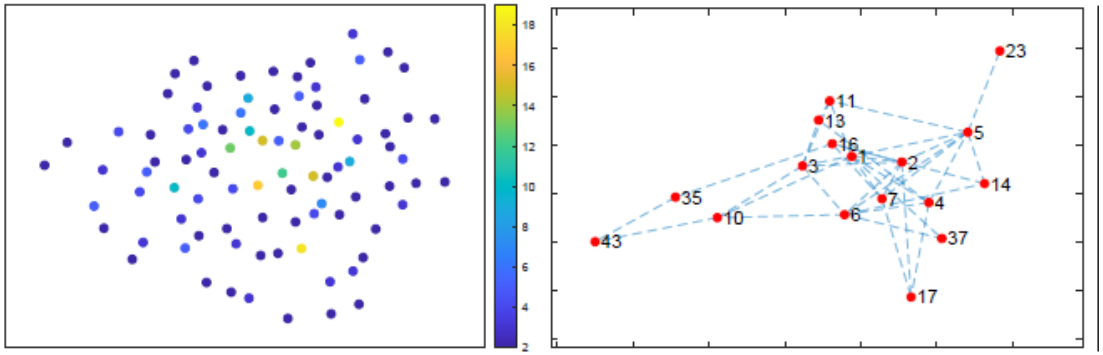
Topo($\|V\|, \|E\|$)	Operation	STSR	TFSR	PL	ND
ISP$(42, 66)$	W/O pooling	0.64	0.68	2.8	3.14
ISP$(12, 18)$	W pooling	0.85	0.84	3.2	3
ISP$(12, 15)$	Pooling, pruning	0.94	0.95	3.8	2.5
ISP$(10, 13)$	Pruning, pooling	0.98	1	3.4	2.6
WS$(100, 200)$	W/O pooling	0.52	0.49	4.2	4
WS$(81, 133)$	W pooling	0.61	0.66	6.7	3.28
WS$(81, 108)$	Pooling, pruning	0.69	0.76	7.1	2.67
WS$(82, 117)$	Pruning, pooling	0.67	0.74	6.9	2.85

Update Cost

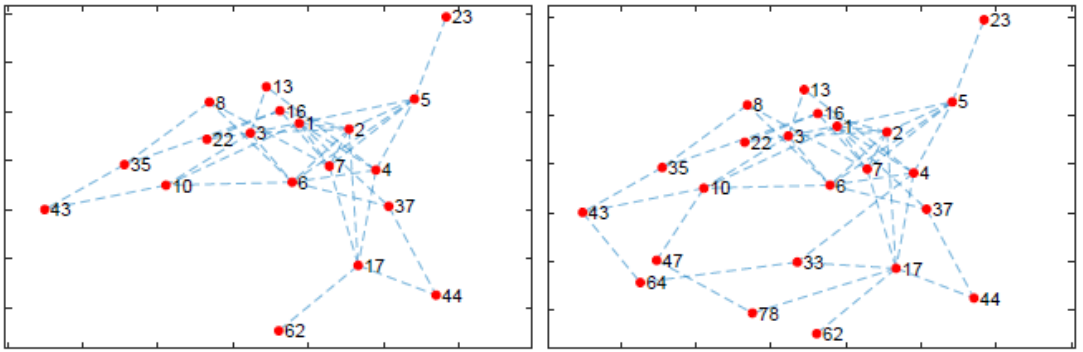
A sample CN:
(V, E) = (90, 199)
Diameter D=8

Another CN:
(V, E) = (70, 197)
Diameter D=4

Global CDS:
Guha/Khuller's solution



(a) The original LN topology. (b) Size of global: 17.



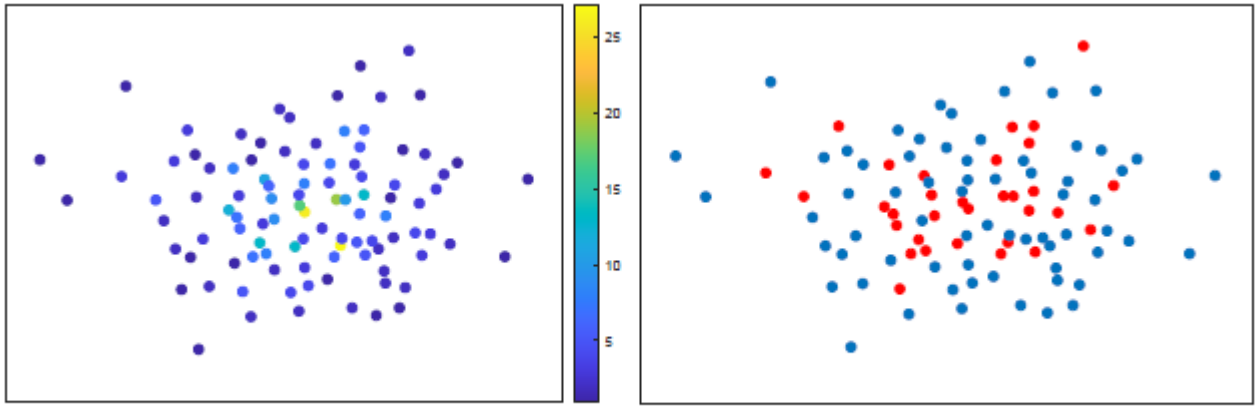
(c) Size of supernodes w/ 3-hop: 19. (d) Size of supernodes w/ 2-hop: 23.

D = 8 / 4	Global	Local	
		3-hop	2-hop
# of supernode	17 / 15	19 / 17	23 / 17
rm-edge	23.4 / 22.7	1.82 / 1.47	1.78 / 1.44
rm-node	24.2 / 24	4.80 / 3.67	4.20 / 3.58
add-edge	23.6 / 22.2	1.96 / 1.33	1.90 / 1.29
add-node	23.9 / 22.6	1.11 / 1.06	1.10 / 1.05
add-node-with-edges	24.0 / 22.3	1.45 / 1.78	1.17 / 1.52

CLoTH Testbed

CLoTH: A payment network testbed, (V, E) = (100, 224)

Transaction #: 1,200 (heterogenous)
Transaction fees: 10 (per node)

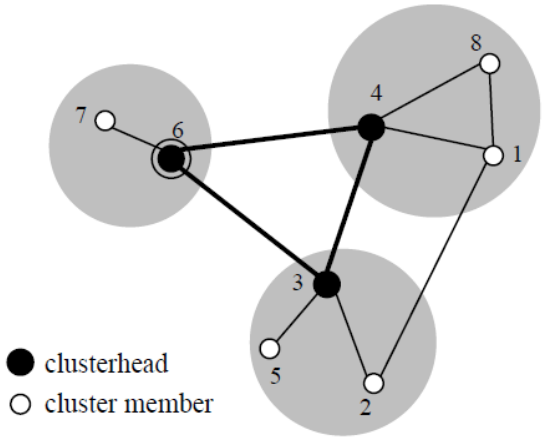


(a) Testbed network. (b) Pooling: supernodes marked red.

TFSR	W/O pooling	W/ pooling	REVIVE	
			every 200 tx	every 400 tx
Random	0.718	0.967	0.932	0.921
Perfect	0.788	0.985	0.941	0.927

6. Future Work

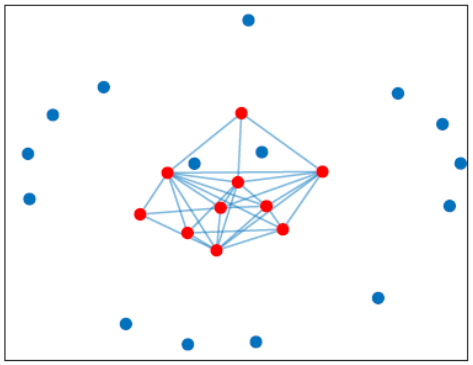
- Hierarchical clustering
  - Supernodes of supernodes (e.g., node 6)

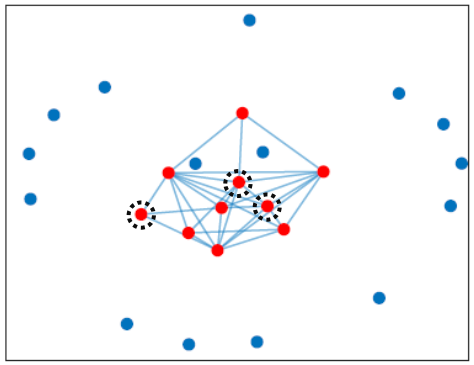


- Trade-offs
  - Benefit (successful transactions)
  - Cost (various fees/updates)

Future Work (Cont'd)

- Impact of locality
  - Value of k on pooling efficiency and local fixes
- ID rotation
  - E.g., ID inversion (for size reduction)





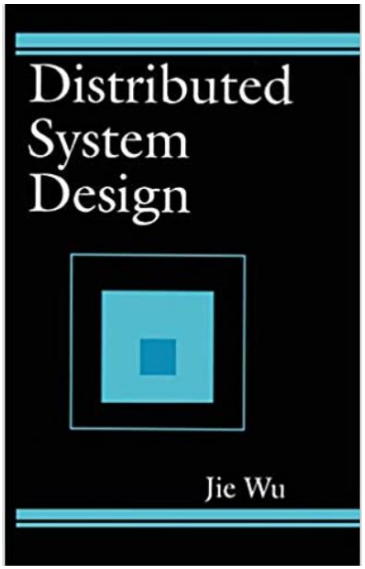
- Others
  - Games: on topology, fund allocation, and routing fees

Future Trends

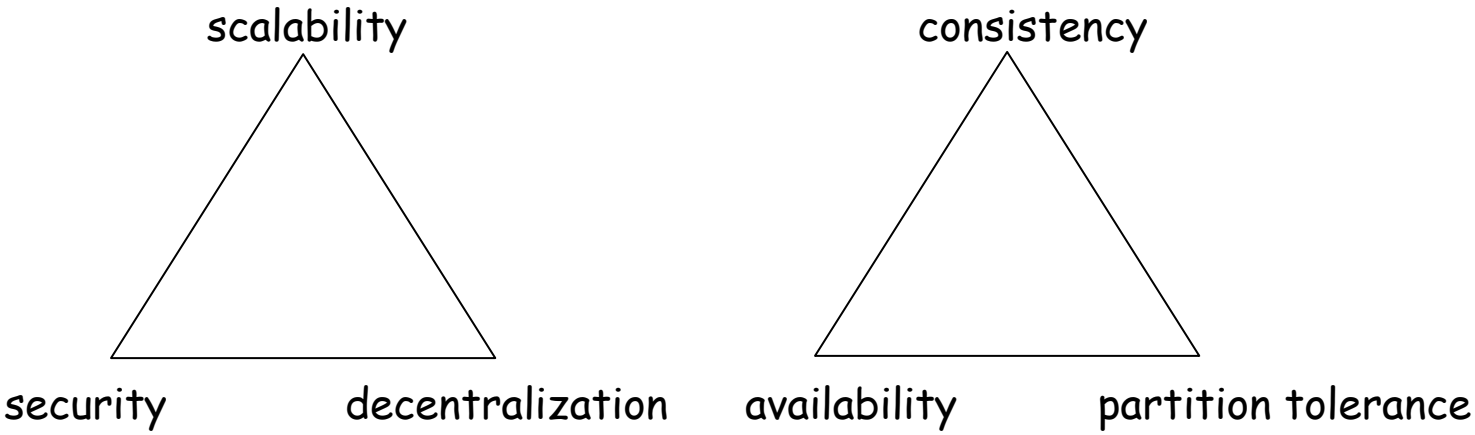
- Future of cryptocurrency
  - Decentralized: Bitcoin
  - Centralized: Digital Currency Electronic Payment (DCEP)
  - In-between: Libra (Novi wallet)
- Blockchain smartphones
  - Commercial: HTC and Samsung
  - Edge blockchain via offloading

Blockchain vs. Distributed Sys. (DS)

- Past results applied in blockchain?
  - Latency hiding
  - Concurrency control
  - Quorum voting



- Trilemma in blockchain and DS



Questions

J. Wu and F. Dai, "*A Generic Distributed Broadcast Scheme in Ad Hoc Wireless Networks*," *IEEE Transactions on Computers*, 2004.

J. Wu and S. Jiang, "Local Pooling of Connected Supernode in Lightning Networks for Blockchains," *Proc. of IEEE Blockchain*, 2020.