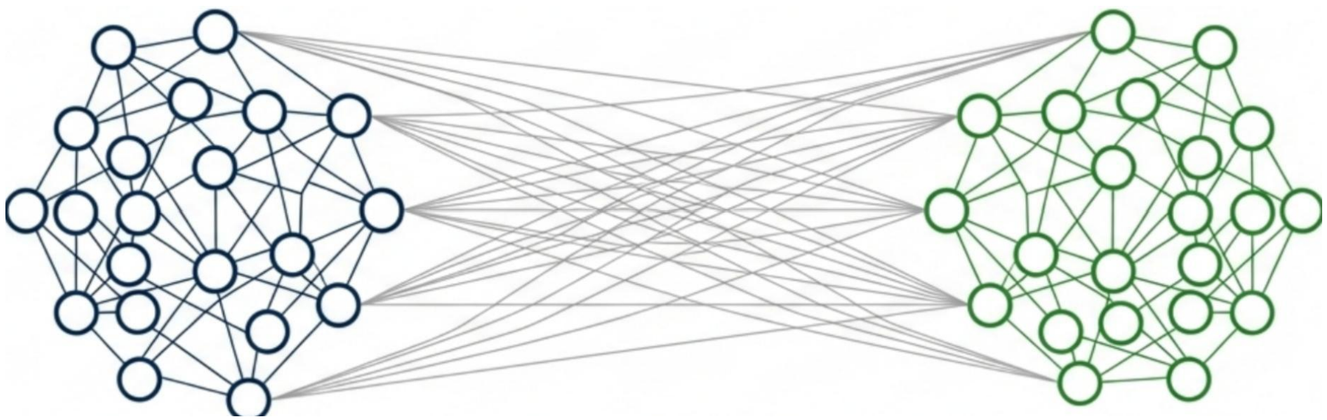
Tsinghua University

Cross-Consensus Reliable Broadcast and its Applications

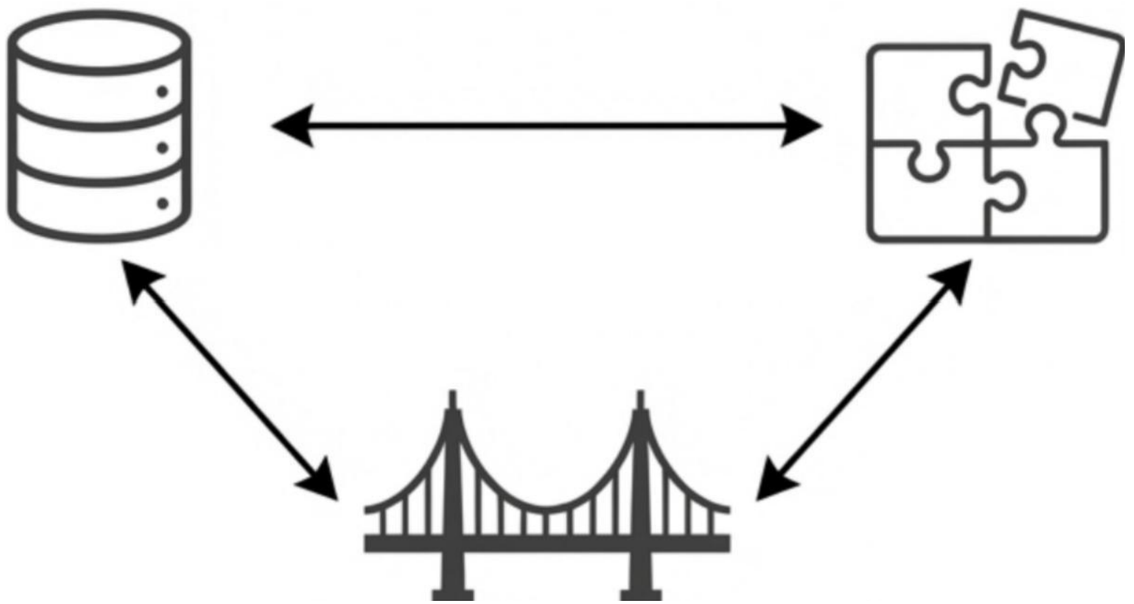


Yue Huang[a], Xin Wang[a], Haibin Zhang[b], Sisi Duan[a,c,d,e]

[a]Tsinghua University, [b]Yangtze Delta Region Institute of Tsinghua University, Zhejiang, [c]Zhongguancun Laboratory, [d]Shandong Institute of Blockchains, [e]State Key Laboratory of Cryptography and Digital Economy Security

The Era of Interconnected Consensus

Replicated State Machines
Client-server coordination across infrastructures



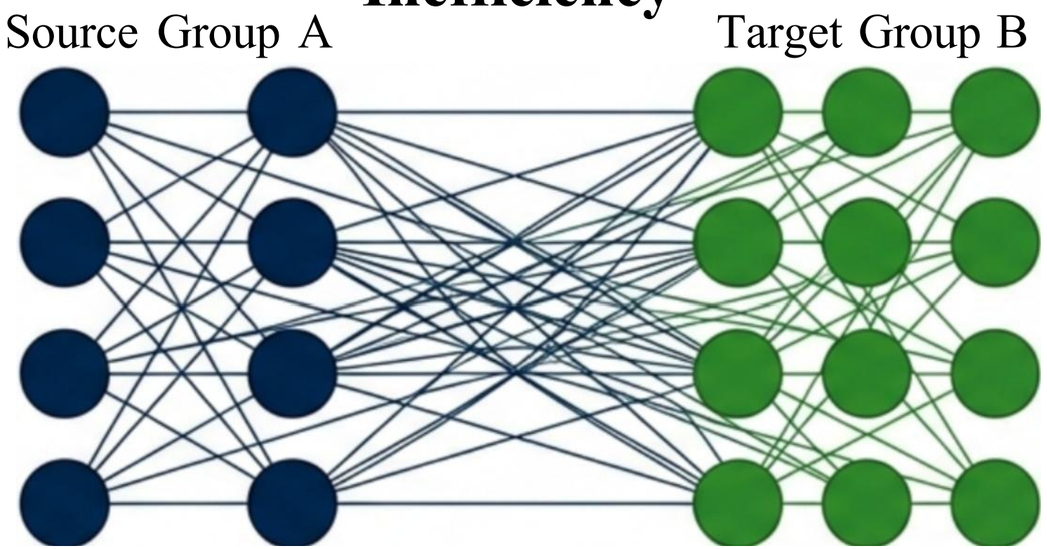
Sharding-Based Blockchains
Cross-shard transaction ordering

Cross-Chain Bridges
Atomicity between heterogeneous chains

Consensus is no longer isolated

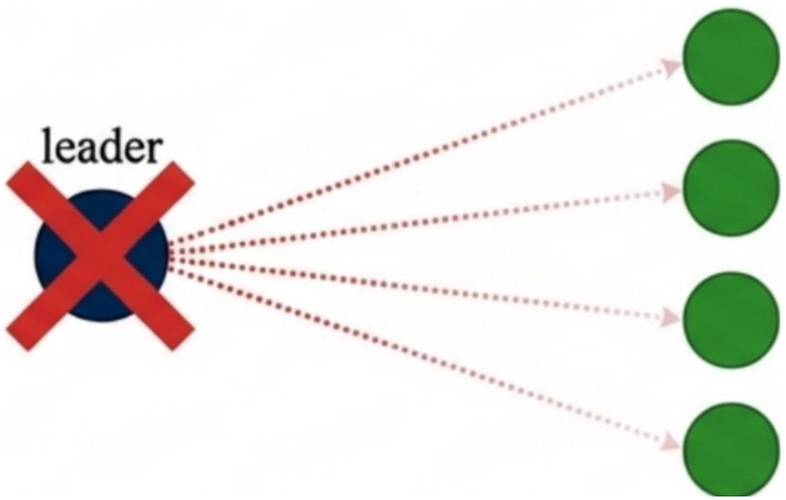
Shortcomings of Prior Works

Inefficiency



$\mathbf{O(mn + n^3)}$ Complexity

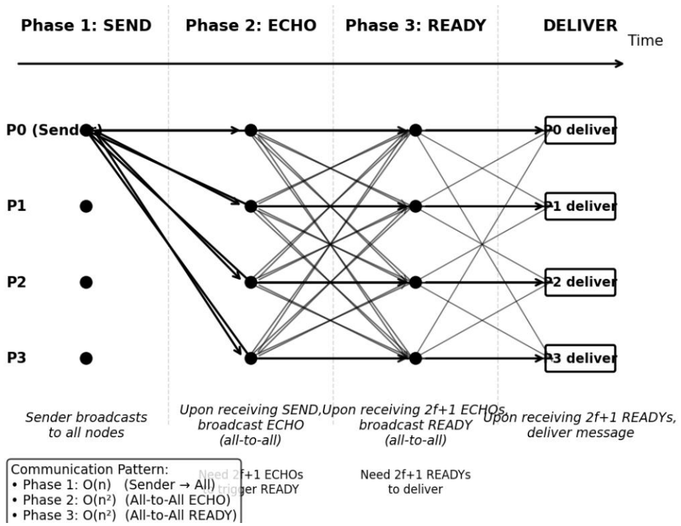
Fragility & Inconsistency



Byzantine Leader Failure

Current solutions treat cross-group communication as an ad-hoc side feature

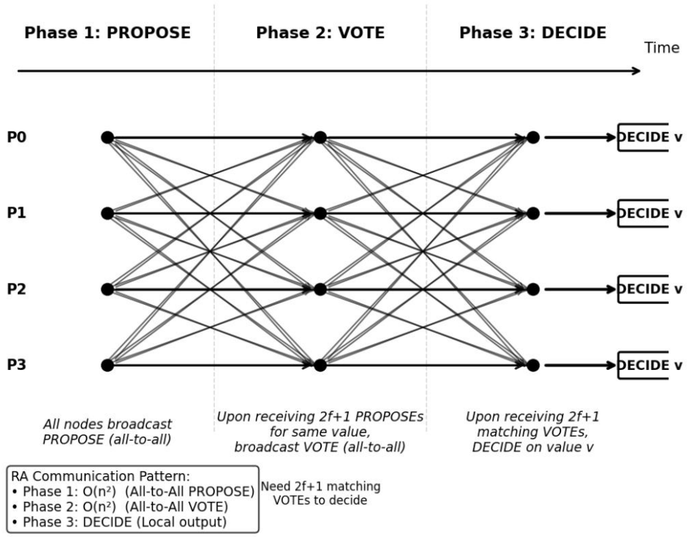
Building Blocks: Primitives



Reliable Broadcast (RBC)

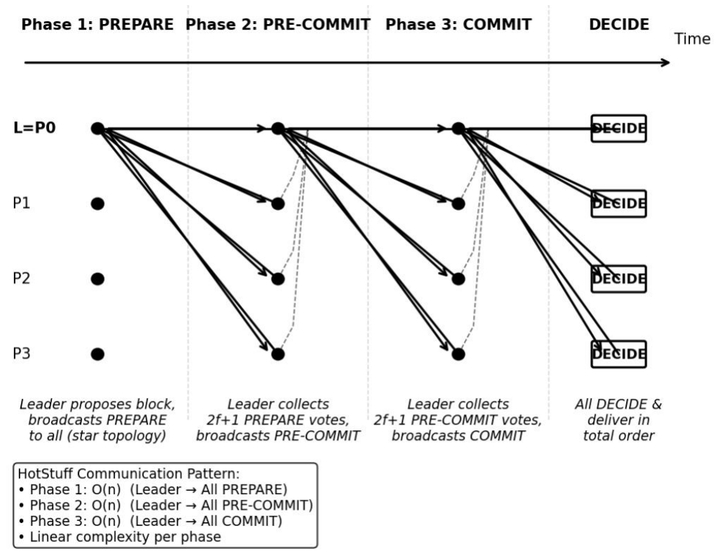
Sender P reliably broadcast a message to a group of nodes

Bracha RBC as an example



Byzantine Reliable Agreement (RA)

All correct nodes reach an agreement on a proposed value, while guaranteeing Validity and Termination



Atomic Broadcast(ABC)

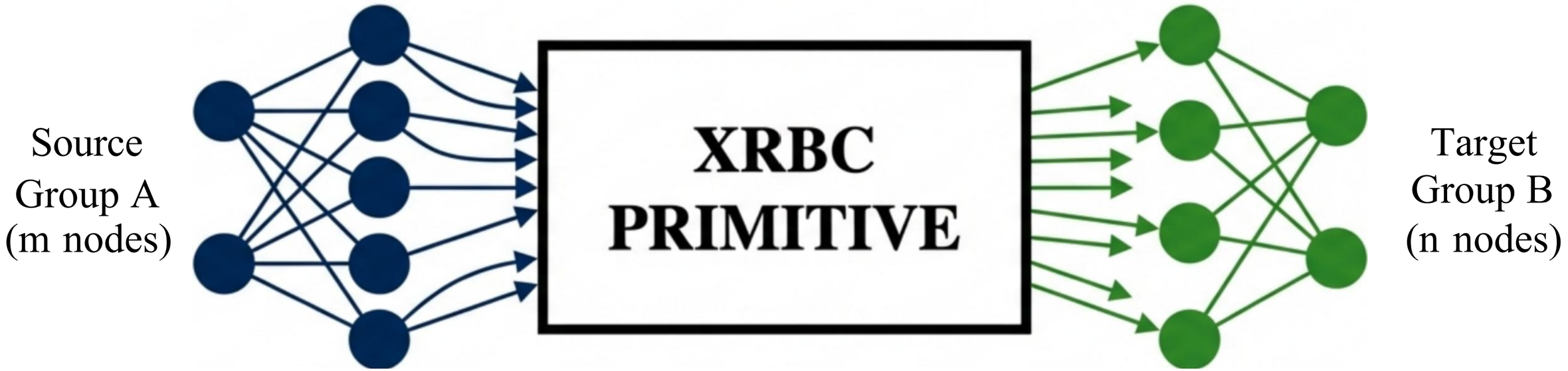
All nodes reach an agreement on the order of messages (values)

Hotstuff as an example

Bracha, Gabriel. "Asynchronous Byzantine agreement protocols." *Information and computation* 75.2 (1987): 130-143.

Yin, Maofan, et al. "HotStuff: BFT consensus with linearity and responsiveness." *Proceedings of the 2019 ACM symposium on principles of distributed computing*. 2019.

The XRBC Primitive



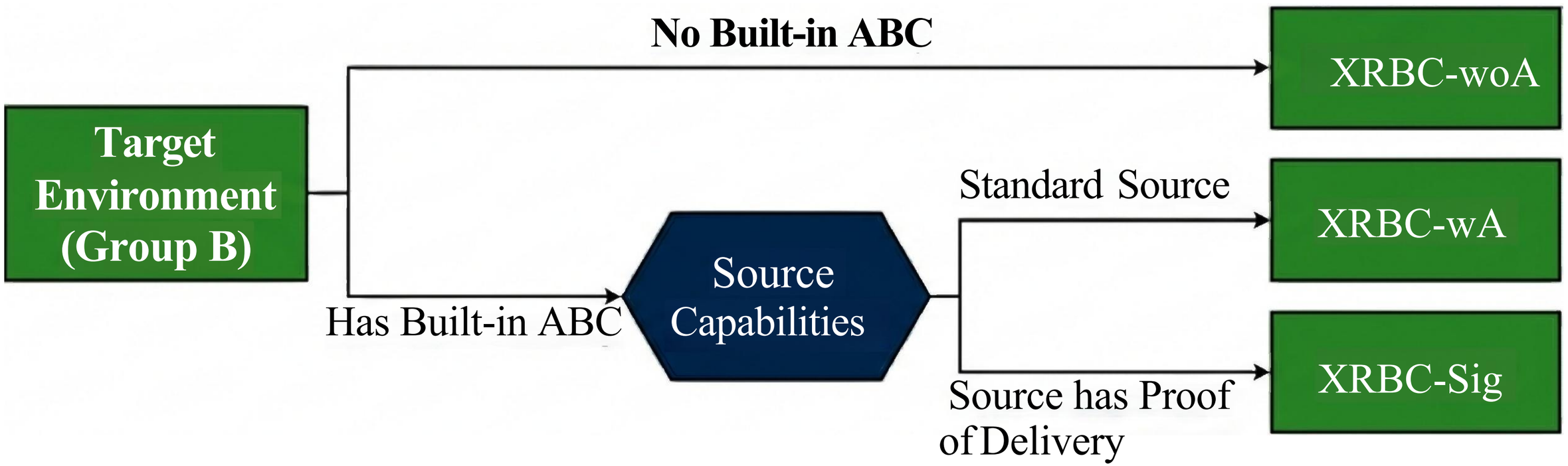
Core Properties:

-**Safety:** If any correct node in B delivers v before v', all do

-**Integrity**: Messages are delivered at most once and must originate from A

-**Termination:** If at least $2t + 1$ correct nodes in A broadcast v, B eventually delivers v

Protocol Constructions Overview



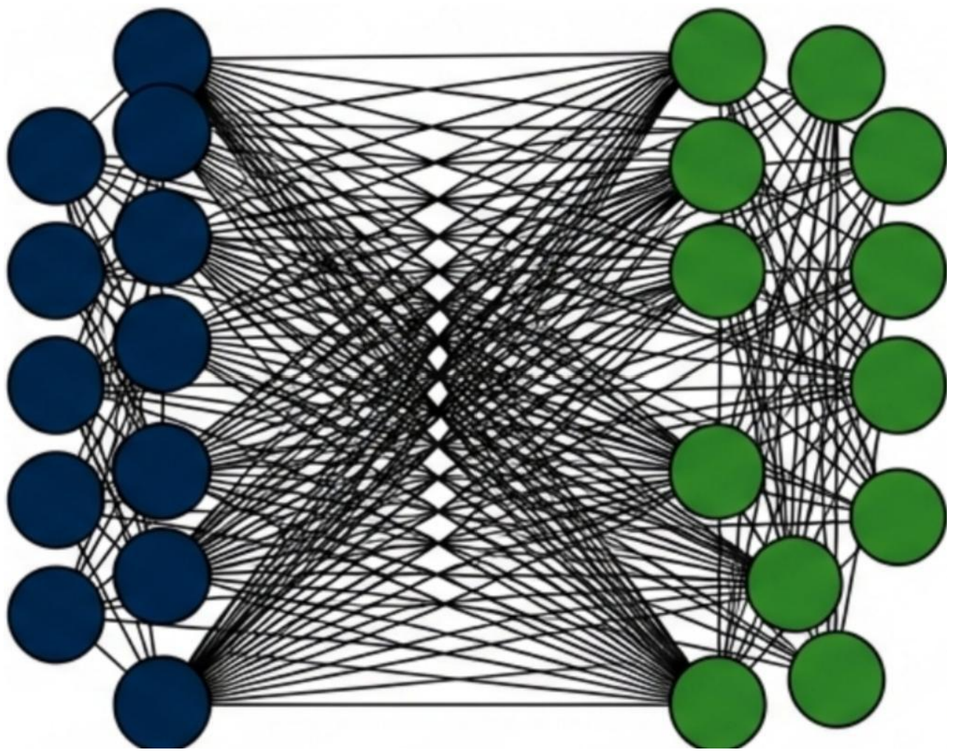
Warm-up: Group-PRBC (Baseline)

Concept: Run m parallel Reliable Broadcast instances

Mechanism: Nodes in B collect vectors of messages and use MVBA to decide

Drawback: High Complexity

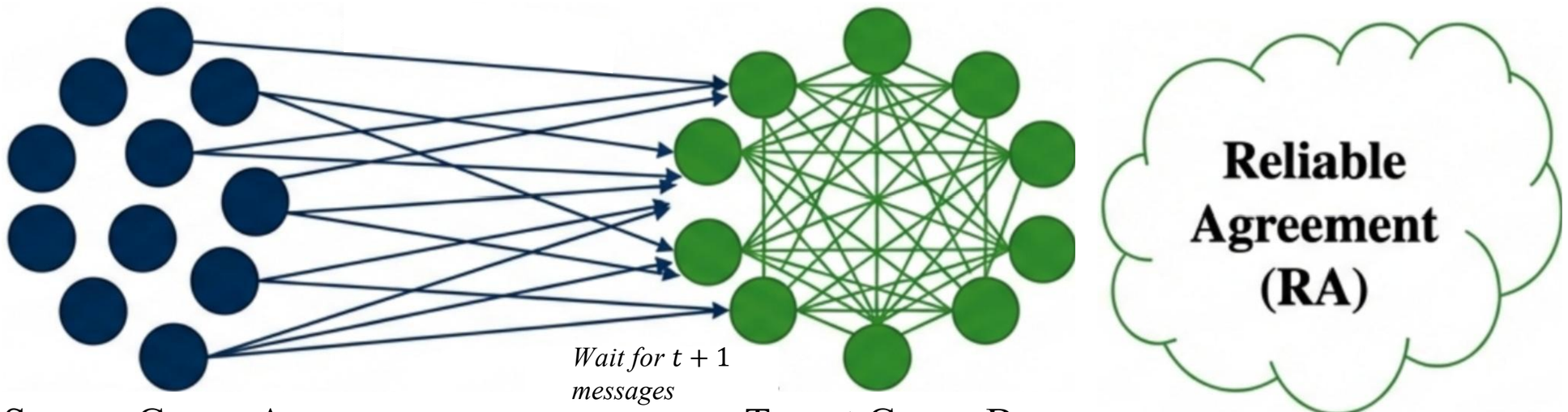
- Message Complexity: $\mathbf{O(mn + n^3)}$
- Inefficient for large groups



Source Group A
(m nodes)

Target Group B
(n nodes)

Protocol 1: XRBC-woA (Without ABC)

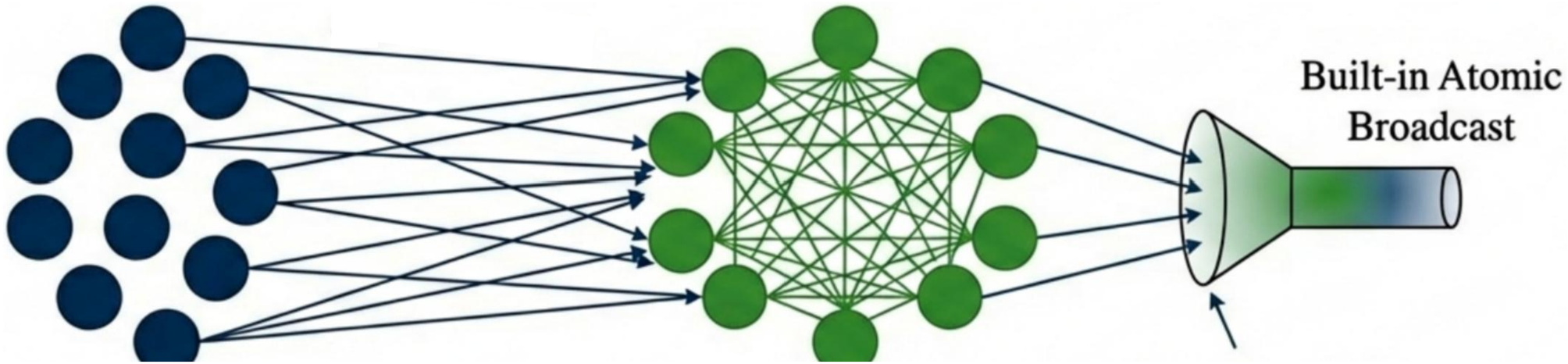


Workflow:

1. Nodes in A send messages to B
2. Nodes in B wait for threshold
3. Nodes in B run Reliable Agreement (RA) to decide

Communication complexity: $\mathbf{O(mnL + \kappa n^2)}$

Protocol 2: XRBC-wA(With ABC)



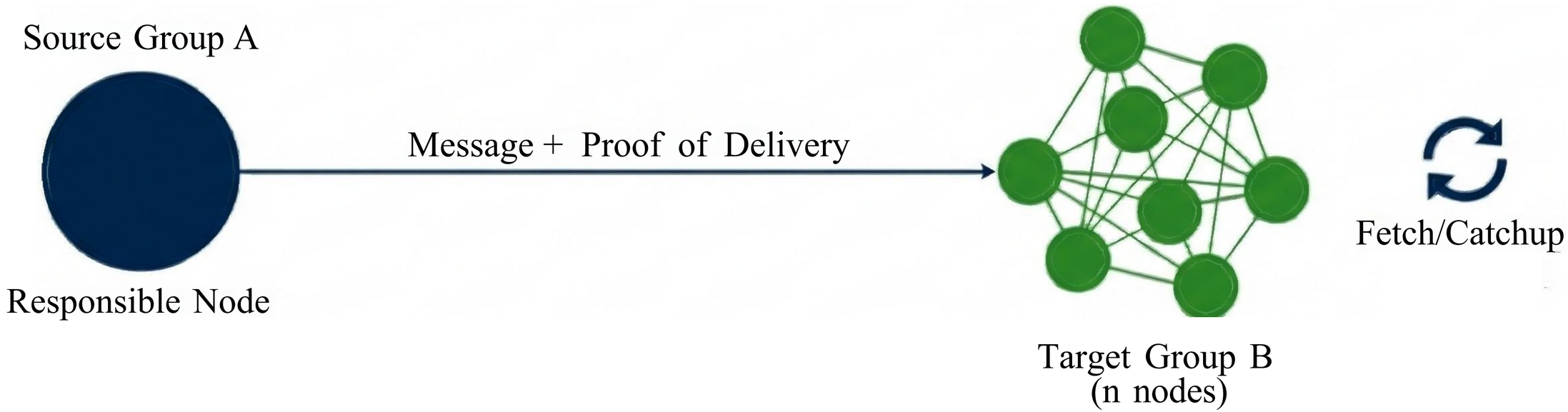
Source Group A
(m nodes)

Target Group B
(n nodes)

Delivery Predicate
(Need $t + 1$ votes)

Optimization: Reuse the target's existing Atomic Broadcast.
Mechanism: Setup a delivery Predicate. Value is valid only if supported by sufficient votes from A.
Benefit: Piggybacks on existing infrastructure
Communication complexity: $\mathbf{O(mnL + C_{ABC}^{\kappa})}$

Protocol 3: XRBC-Sig (Signatures)



Target: High-efficiency environments supporting Proof of Delivery

Safety Net: Fetch/Catchup mechanism handles Byzantine Responsible Nodes

Communication complexity: $\mathbf{O(nL + \kappa n + C^{\kappa}_{ABC})}$. Decoupled from source size m

Complexity Comparison

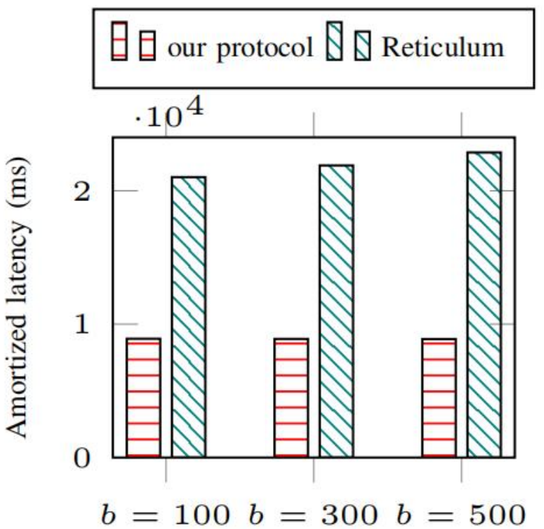
Protocol	Assumption	Messages	Communication	Time
Group-PRBC (Baseline)	trusted PKI	$O(mn + n^3)$	$O(nmL + \kappa nm + \kappa n^2 + mn^2 \log n)$	$O(1)$
Group-PRBC (Baseline)	none	$O(mn + n^3)$	$O(mnL + \kappa n^2 m + \kappa n^3)$	$O(1)$
XRBC-woA	ABC in A; hash	$O(mn + n^2)$	$O(mnL + \kappa n^2)$	$O(1)$
XRBC-wA	ABC in A and B; hash	$O(mn + M_{ABC})$	$O(mnL + C_{ABC}^{\kappa})$	$O(T_{ABC})$
XRBC-Sig	proof of delivery in A; hash; ABC in A and B; trusted PKI	$O(n + M_{ABC})$	$O(nL + \kappa n + C_{ABC}^{\kappa})$	$O(T_{ABC})$

*m is the size of group A and n is the size of group B. L is the length of the cross-group/consensus messages. κ is the length of the security parameter (e.g., length of a digital signatureor a hash). $O(T_{ABC})$ and $O(M_{ABC})$ are the time complexity and message complexity of ABC in B, respectively. $O(C_{ABC}^{\kappa})$ is the communication complexity of ABC for κ-bit inputs.

Application I: Cross-Shard Coordination (Reticulum)

Reticulum (NDSS '24) uses a control shard to coordinate process shards.

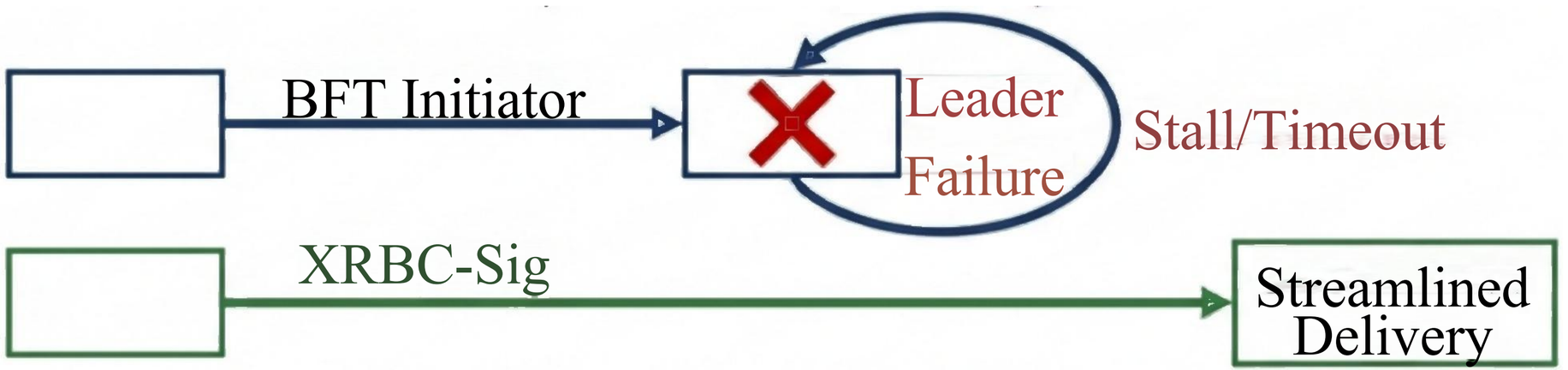
- Problem: The $(\Delta + \delta)$ −BB protocol is too expensive
- Communication complexcity: $\mathbf{O(\kappa(m + n)^3)}$ under the trusted setup assumption
- Result: **57.03%-61.16% lower latency** for $f = 30$ compared to the vanilla approach.



(c) Latency of our protocol and Reticulum for $t = 1$ and $f = 30$.

Xu, Yibin, et al, "A Two-Layer Blockchain Sharding Protocol Leveraging Safety and Liveness for Enhanced Performance.", in *NDSS,* 2024.

Application II: Cross-Shard Transactions (Chainspace)



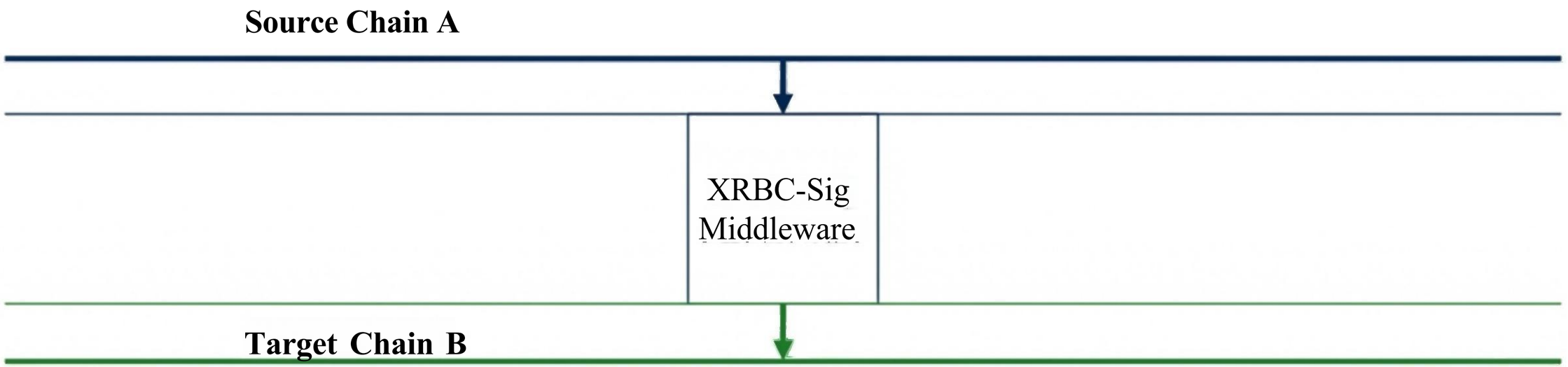
Chainspace (NDSS'18) use a "BFT-Initiator" to notify local decisions

•Problem: BFT-Initiator optimization is fragile. Leader failure cause complex timeout timer to recover

•Result: **Reduces latency by 48.4%-52.0%** and eliminates complex manual timeout logic

M. Al-Bassam, A. Sonnino, S. Bano, D. Hrycyszyn, and G. Danezis, "Chainspace: A sharded smart contracts platform", in *NDSS*, 2018.

Application III: Cross-Chain Bridge



Using **XRBC-Sig as Middleware** transferring assets between Source Chain A and Target Chain B.

Guarantee: The Termination property ensures atomicity (if it happens on A, it will happen on B).

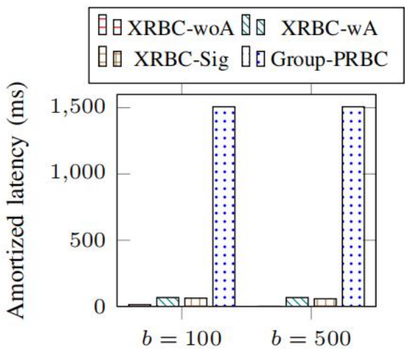
Evaluation: Scale and Speed

Setup: AWS EC2, WAN environment

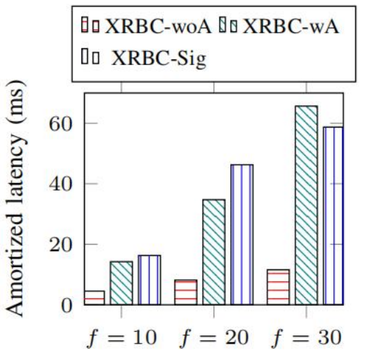
Latency improvement: compared to baseline, the latency of our 3 XRBC is much better

Scalability: Tested fault tolerance f=10 to 30, latency degradation is minimal and linear

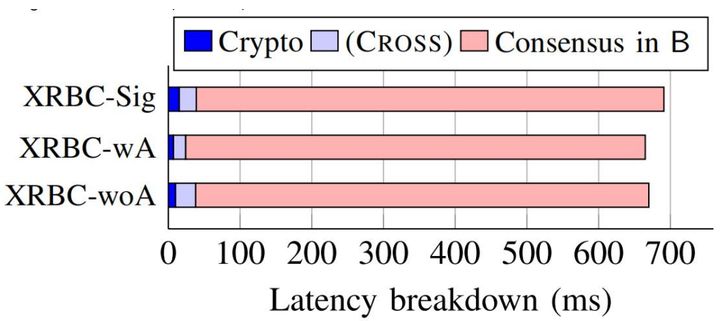
Bottleneck analysis: The consensus in the Target Group B dominates latency. The XRBC overhead is negligible



(a) Latency of XRBC and Group-PRBC for $t = 1$ and $f = 30$.



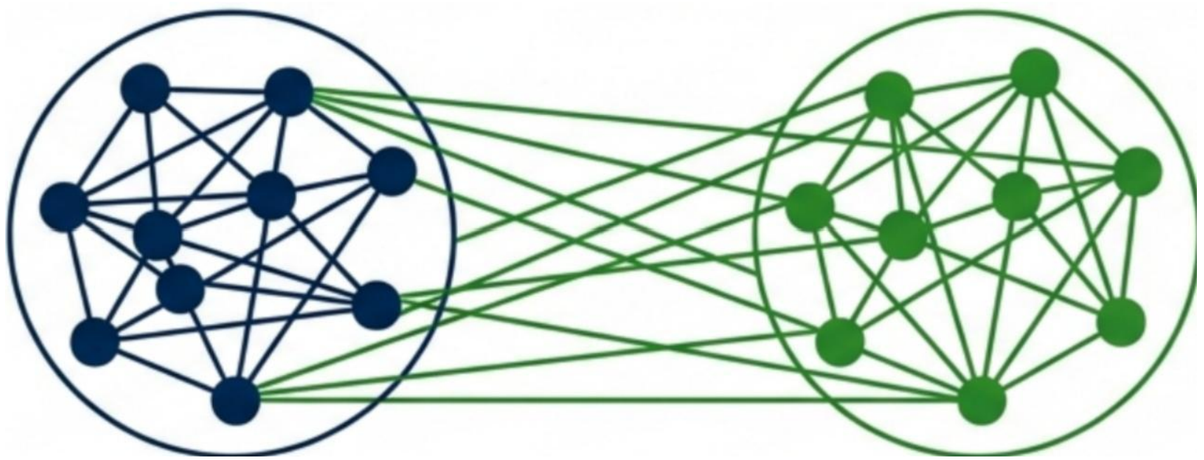
(c) Amortized Latency of XRBC-woA, XRBC-wA, and XRBC-Sig for $t = 1$ and $f = 10, 20, 30$.



(f) Latency breakdown of XRBC-woA, XRBC-wA, and XRBC-Sig for $t = 1$ and $f = 30$.

Conclusions

Source Group A



Target Group B

Contributions: formalize the notion of XRBC; can be directly use or extended to benefit cross-shard coordination, cross-shard transactions and cross-chain bridges

Toolkit: Three constructions (XRBC-woA, wA, Sig) tailored to different assumptions

Foundation of future works: XRBC offers a secure and practical foundation for the next generation of multi-consensus systems