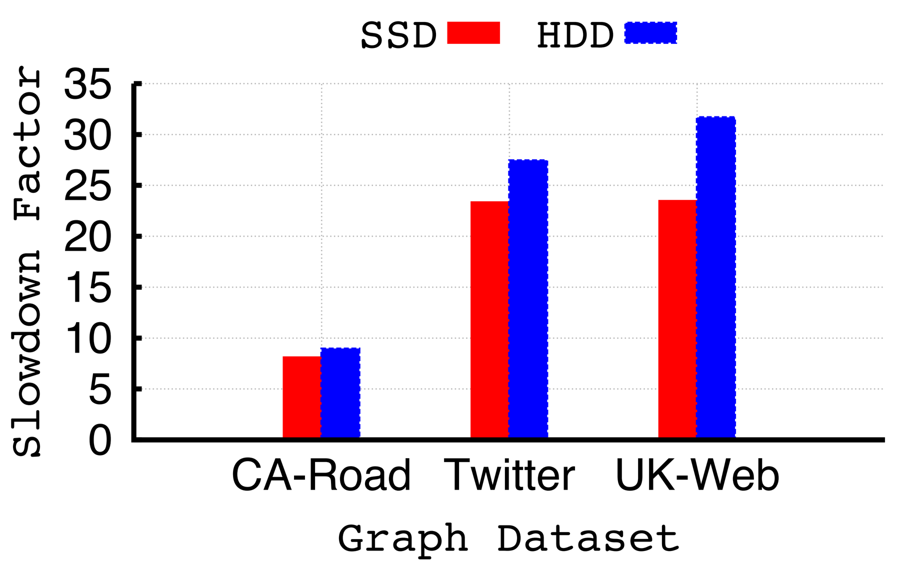
Zorro: Zero-Cost Reactive Failure Recovery in Distributed Graph Processing

Mayank Pundir, Luke M. Leslie, Indranil Gupta, Roy H. Campbell
University of Illinois at Urbana-Champaign

Problem

Current failure recovery mechanisms (such as checkpointing) are **proactive**:

- Incur high and unnecessary **overhead** during normal, failure-free execution.



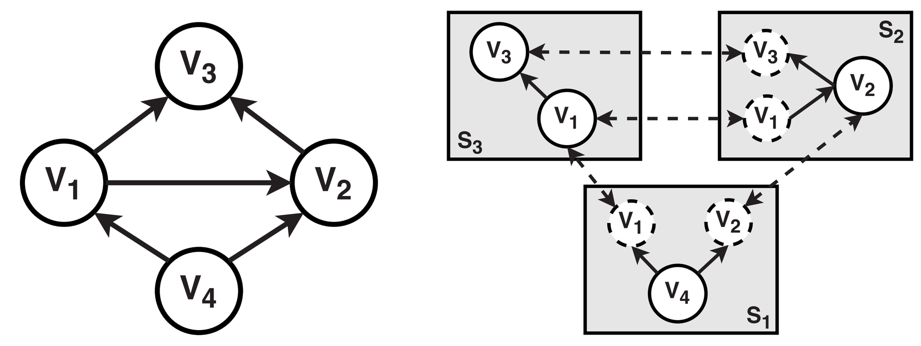
Checkpointed iteration slowdown in PowerGraph

- Need to **repeat computation** between the last checkpoint and the time of failure.
- Imposes a **complex trade-off**: Frequent checkpoints produce waste, infrequent checkpoints produce risk.

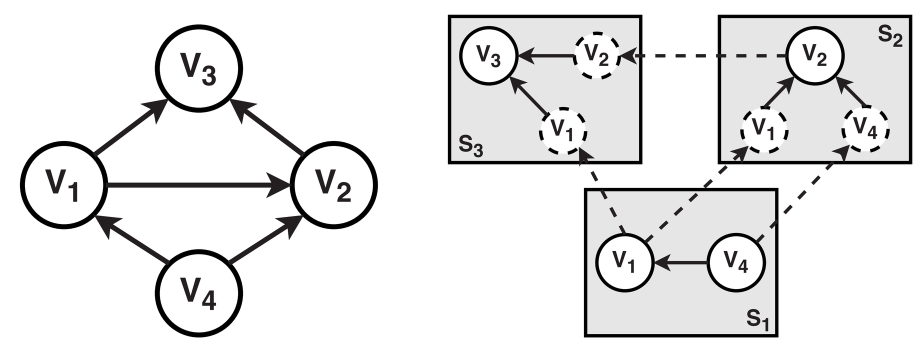
Natural Replication

Communication between partitions introduces **state replication** at remote servers:

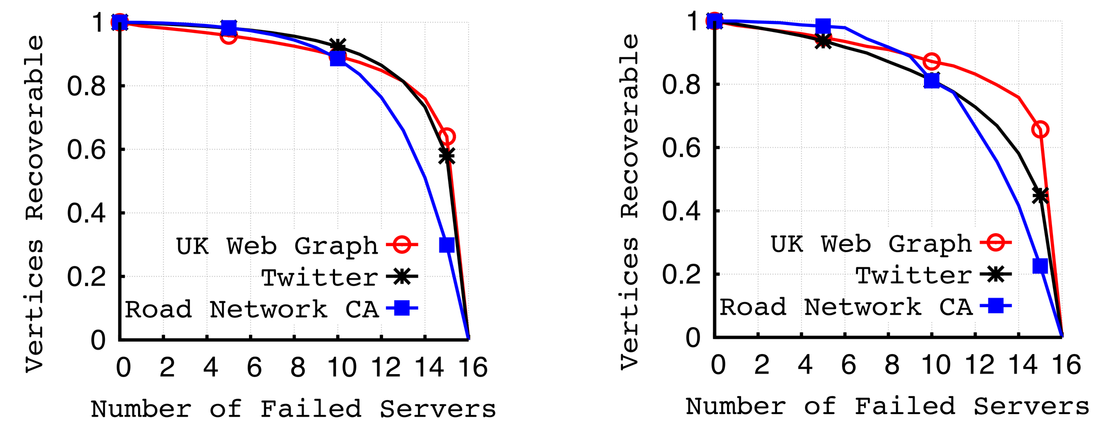
- **All-Neighbor Replication** (e.g., PowerGraph): Vertex states replicated at all neighbors on remote servers.



- **Out-Neighbor Replication** (e.g., Pregel, Giraph, LFGraph): Vertex states replicated at out-neighbors on remote servers.



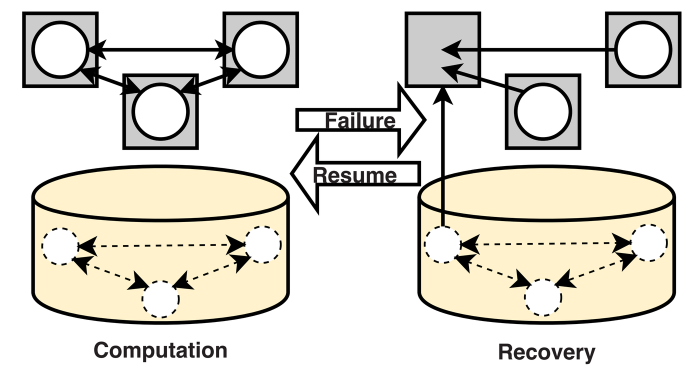
Reactive Recovery



PowerGraph **LFGraph**

State recoverability using existing replication.

- **99%** state recoverable when 6-12% servers fail, and **87-95%** when half the cluster fails.
- Reactive recovery incurs **zero-cost** during failure-free execution, and performs **opportunistic retrieval** of replicated vertices.



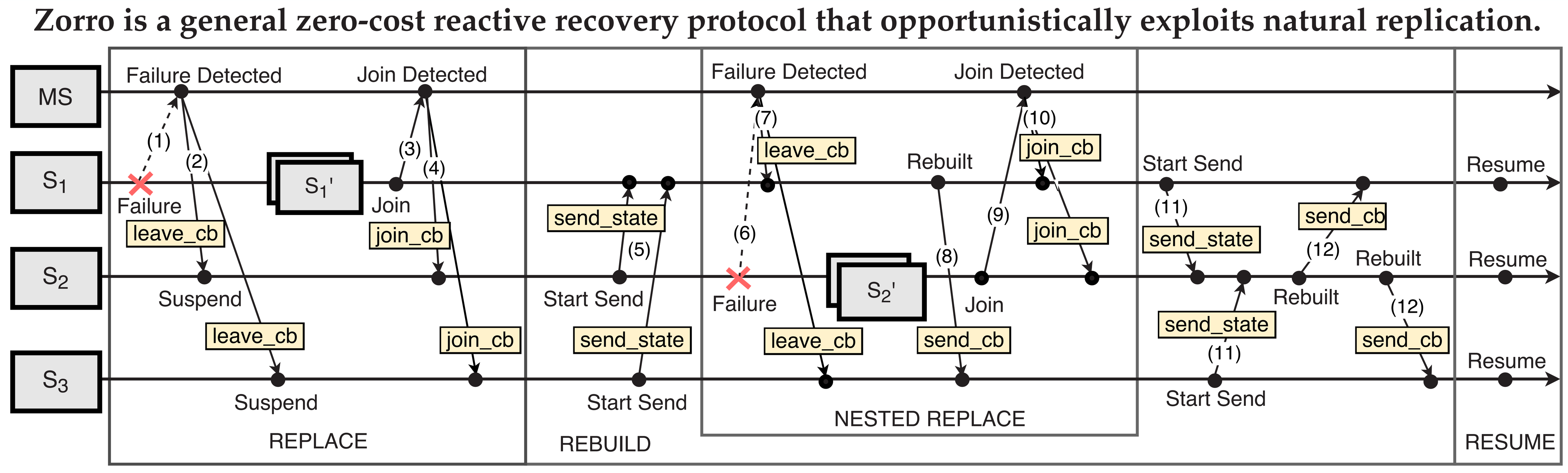
Desired Characteristics

- **ZO** Zero overhead incurred during normal, failure-free execution.
- **CR** Complete recovery of the graph state from just before the point of failure.
- **FR** Fast recovery after failures.

Traditional proactive approaches prioritize CR. Zorro strives for ZO+FR+near-CR.

Three R's of Zorr(r)o

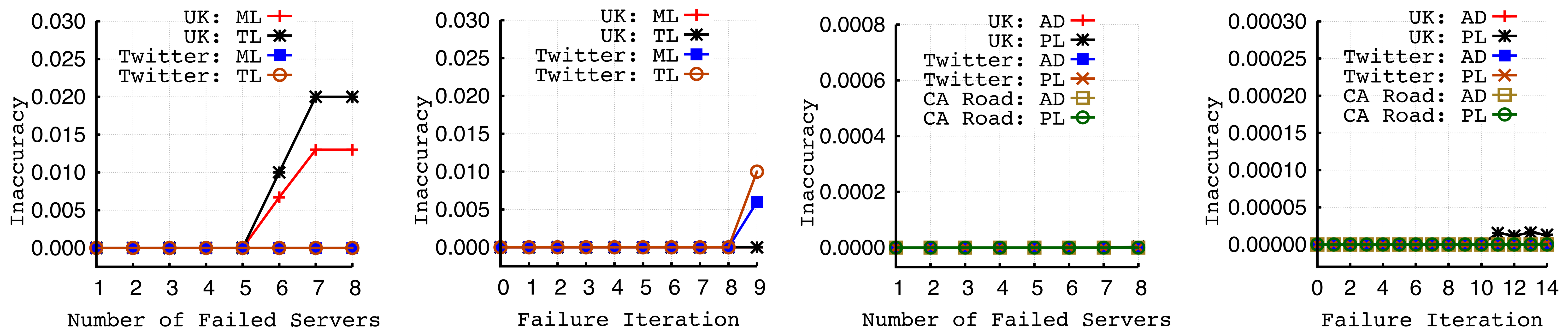
Zorro is a general zero-cost reactive recovery protocol that opportunistically exploits natural replication.



Replace failed servers (using, e.g., ZooKeeper). Rebuild state of replacement servers using available state replicas. Resume from the start of failed iteration, after partial scatter.

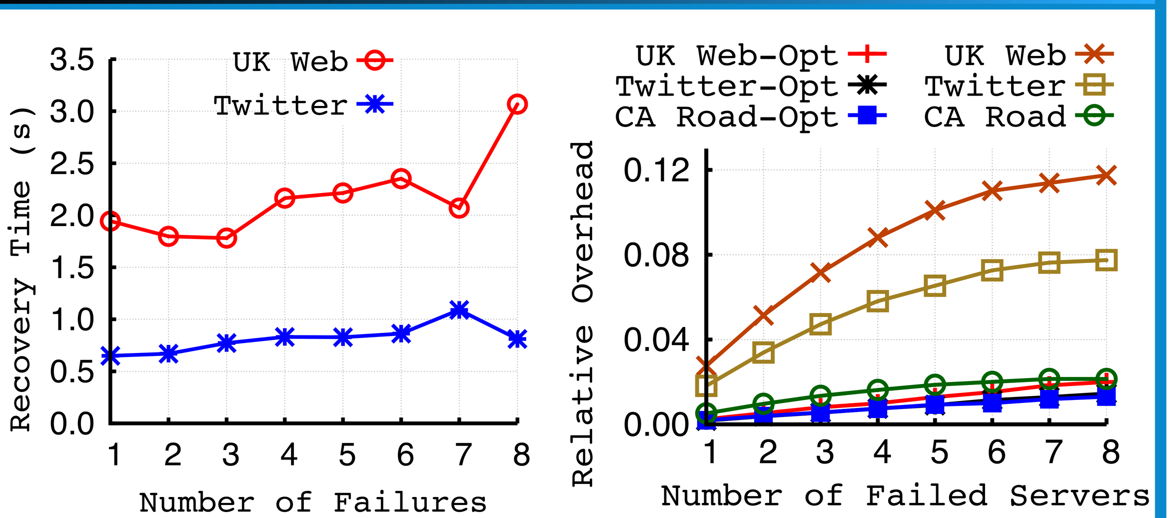
Accuracy Evaluation

Application-specific metrics. **PageRank:** • *ML*: PageRank mass lost • *TL*: PageRank top-k lost. **SSSP:** • *PL*: paths lost • *AD*: average normalized path difference.



PageRank in PowerGraph **SSSP in PowerGraph**

Recovery Overhead



Recovery Time **Network Overhead**

Recovery time and network overhead are a fraction of that in a normal iteration.

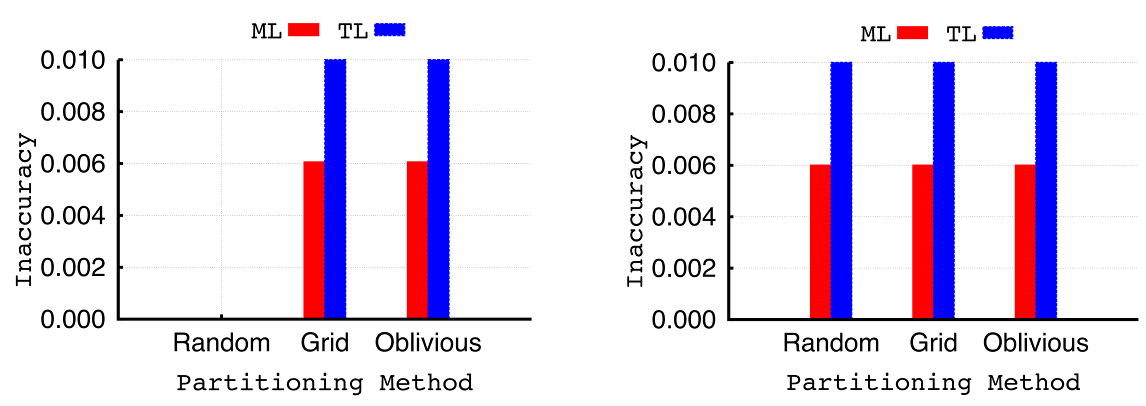
Funding and Research Groups

This work was supported in part by NSF and AFOSR/AFRL grants.

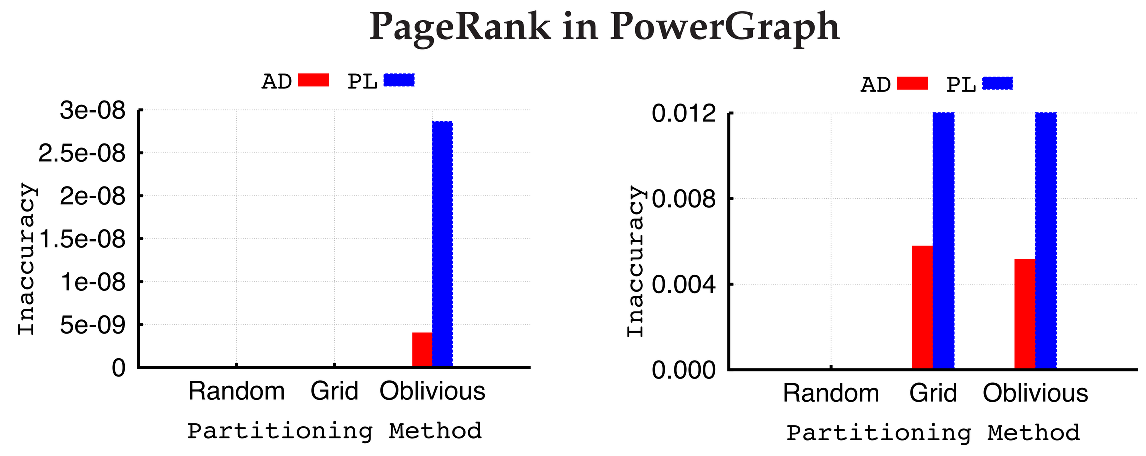
Distributed Protocols Research Group: http://dprg.cs.uiuc.edu/

Systems Research Group: http://srg.cs.illinois.edu/

Partitioning Methods



PageRank in PowerGraph



SSSP in PowerGraph

Intelligent partitioning results in *at most* 1.2% increase in inaccuracy.

Summary

- **Accurate:** Zero inaccuracy in many scenarios, worst-cast accuracy of 97%.
- **Fast:** Recovery costs less than a single iteration, specific optimizations available.
- **Scalable:** Both the expected number of recovered vertices and the probability of recovery (assuming hash partitioning) depend only on the fraction of failed servers.
- **Resilient:** Rebuild of each failed server is *independent and concurrent* across survivors, easily handling cascading failures.