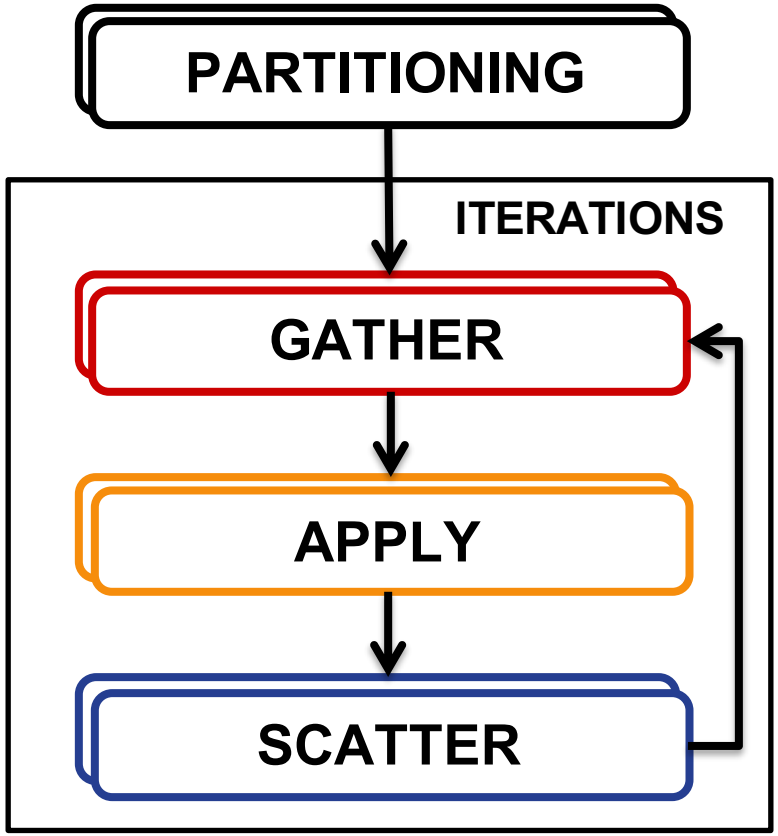
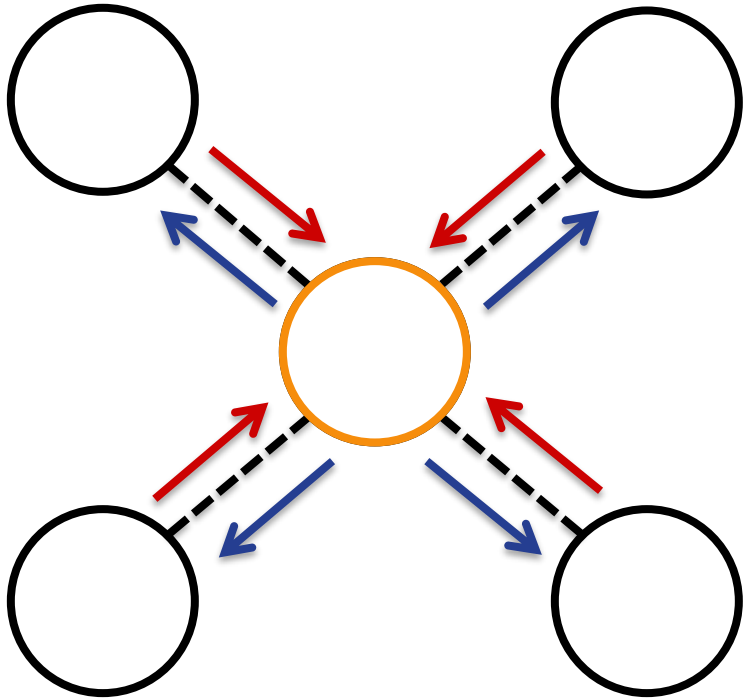
Zorro: Zero-Cost Reactive Failure Recovery in Distributed Graph Processing

Mayank Pundir*, Luke M. Leslie, Indranil Gupta, Roy H. Campbell
University of Illinois at Urbana-Champaign
*Facebook (work done at UIUC)

Synchronous Gather-Apply-Scatter





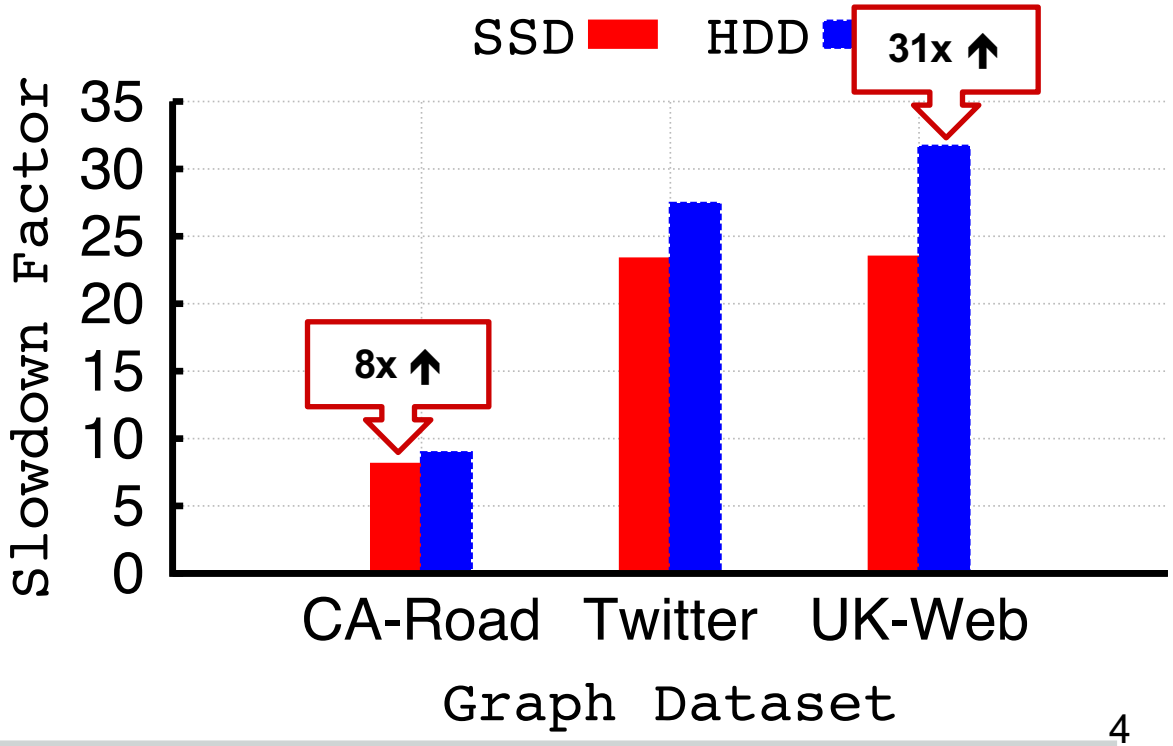
Checkpointing

- Proactively save state to persistent storage
- Used by:
  - **PowerGraph** [Gonzalez et al. OSDI 2012]
  - **Giraph** [Apache Giraph]
  - Distributed GraphLab [Low et al. VLDB 2012]
  - Hama [Seo et al. CloudCom 2010]

Checkpointing is Expensive

8 – 31x Increased Per-Iteration Execution Time

Graph Dataset	Vertex Count	Edge Count
CA-Road	1.96 M	2.77 M
Twitter	41.65 M	1.47 B
UK Web	105.9 M	3.74 B



Checkpointing is Flawed

- After failure, redoing iterations takes time, and adds to the run time
- Checkpointing is hard to configure:
  - If high checkpointing interval, a checkpoint may not even be available
  - If low, checkpoints may be wasted

Failures are not that common

- 9 failures per every 100 servers among over 100,000 servers studied over 14 months - Vishwanath et al., (Microsoft Research), SoCC 2010.

- 1000 individual server failures, 20 rack failures among other failures in first year for a new cluster containing thousands of machines - Jeff Dean (Google), SoCC 2010 keynote.

Checkpointing is Flawed

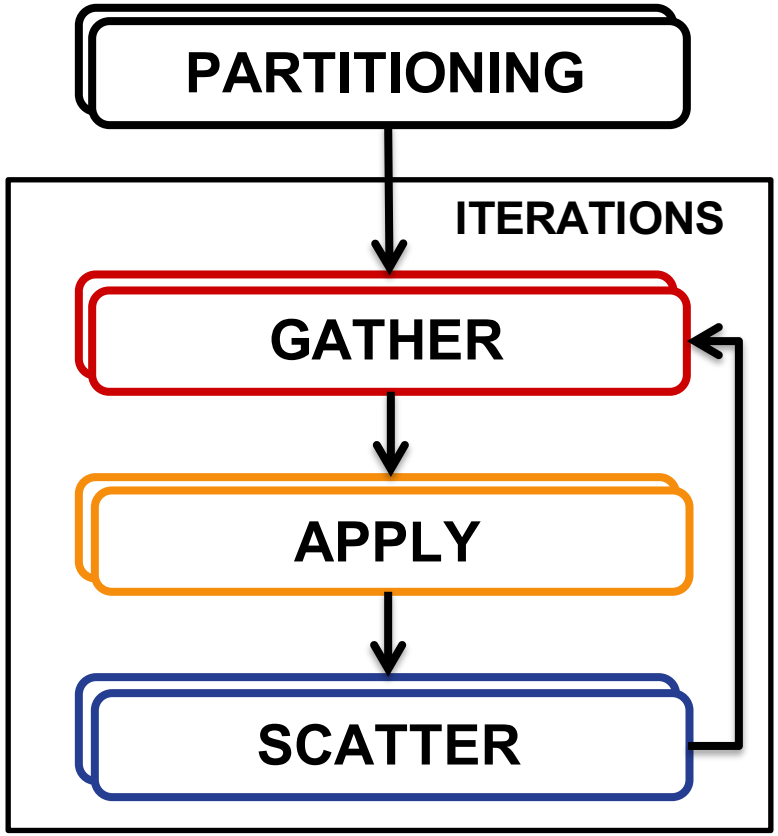
- "While we could turn on checkpointing to handle some of these failures, in practice we choose to disable checkpointing." [Ching et. al. (**Giraph** @ Facebook) VLDB 2015]
- "Existing graph systems only support checkpoint-based fault tolerance, which most users leave disabled due to performance overhead." [Gonzalez et. al. (**GraphX**) OSDI 2014]
- "The choice of interval must balance the cost of constructing the checkpoint with the computation lost since the last checkpoint in the event of a failure." [Low et. al. (**GraphLab**) VLDB 2012]
- "Better performance can be obtained by balancing fault tolerance costs against that of a job restart." [Low et al. (**GraphLab**) VLDB 2012]

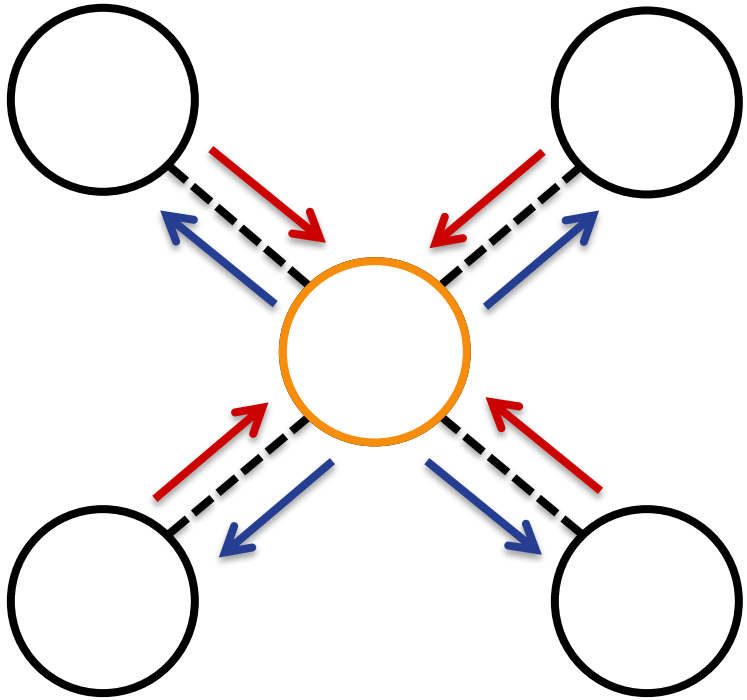
Alternatives to Checkpointing

- Restarting computation may be expensive – production jobs may take as much as an hour [Ching et. al. (Facebook) VLDB 2015]

- **Can we do better? Can we disable checkpointing altogether and still recover from failures?**

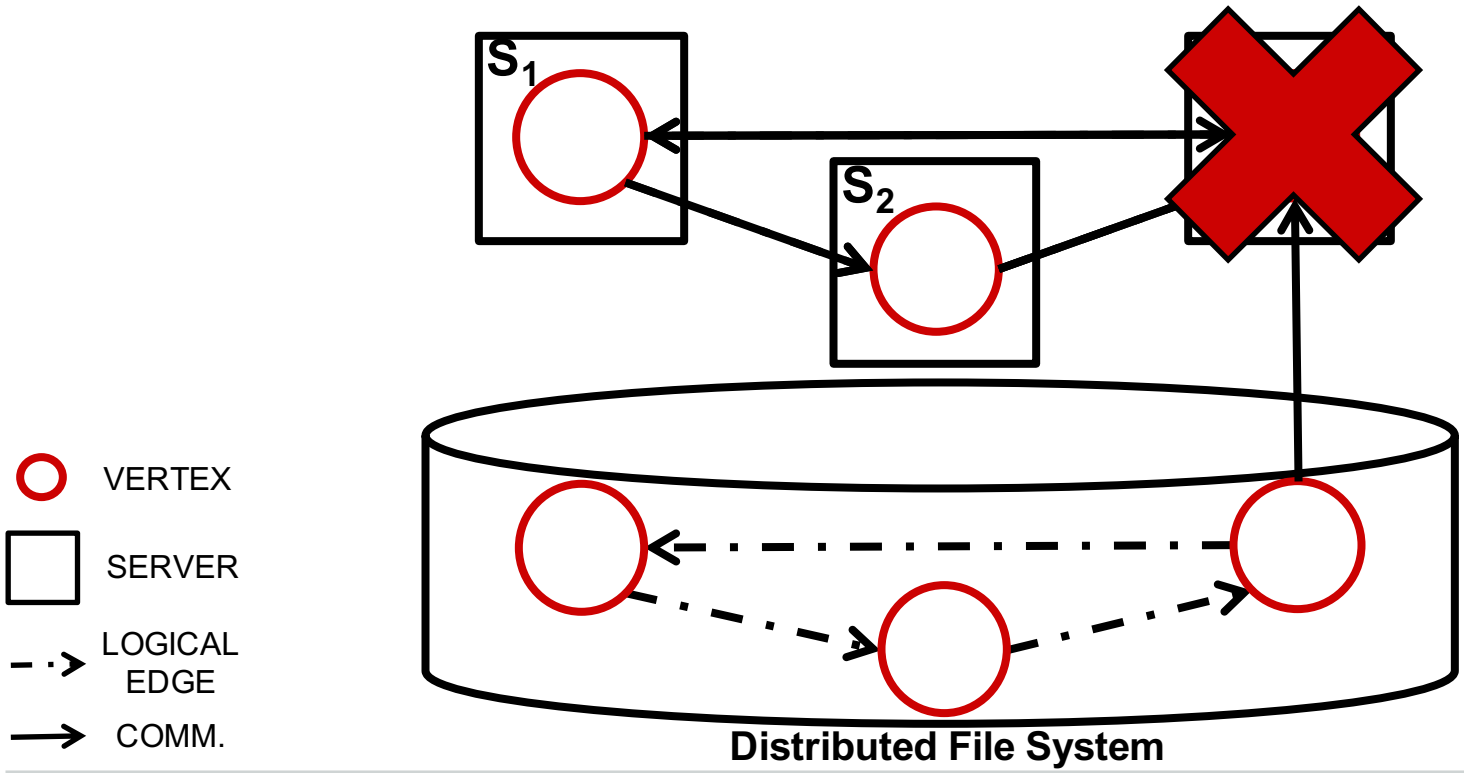
Scatter Leads to Replication





Zero-cost, Reactive Recovery

Recovery using natural replication



Key Questions

- How does natural replication occur in distributed graph processing systems?
- How much graph state is recoverable using the natural replication?
- How much application accuracy is achievable by relying on natural replication alone?

Natural Replication

Out-neighbor Replication

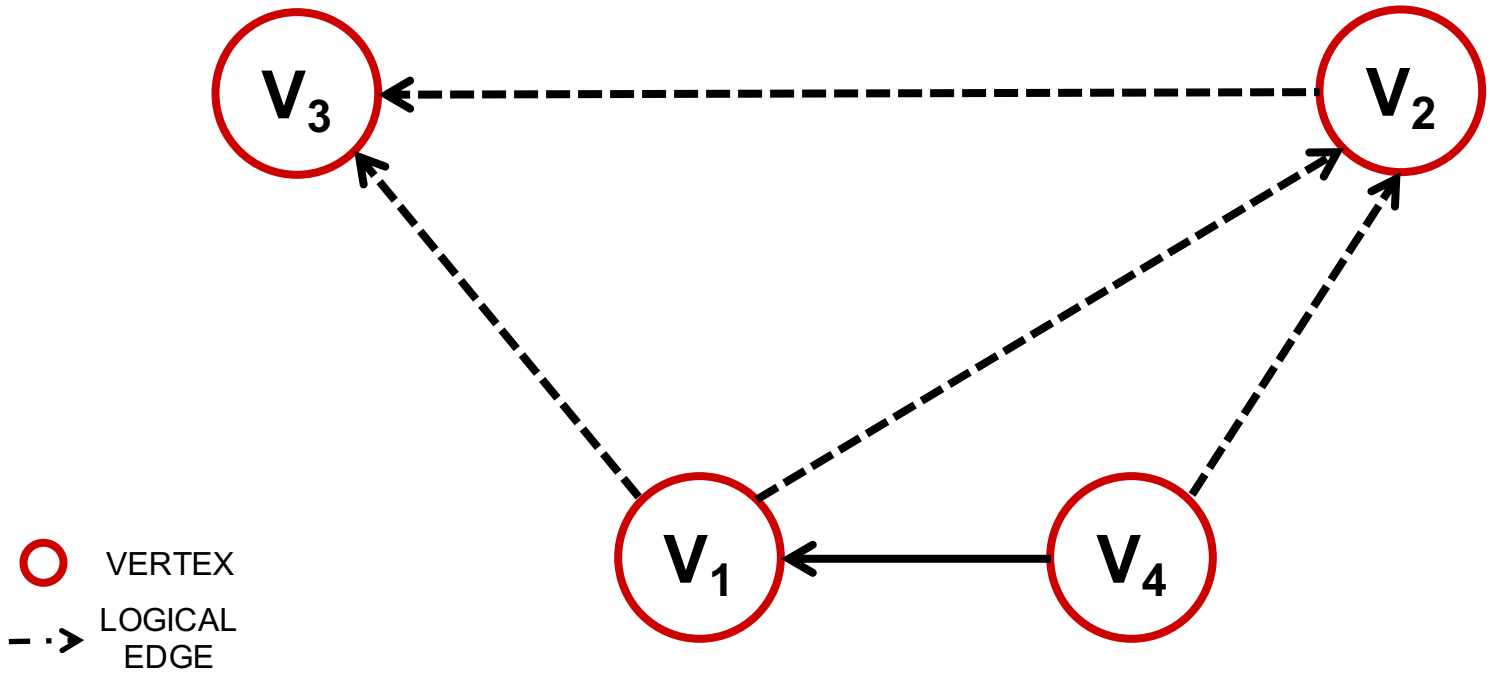
- Created by vertex partitioning
- LFGraph, Giraph (old), Hama

All-neighbor Replication

- Created by edge partitioning
- PowerGraph, Giraph (new)

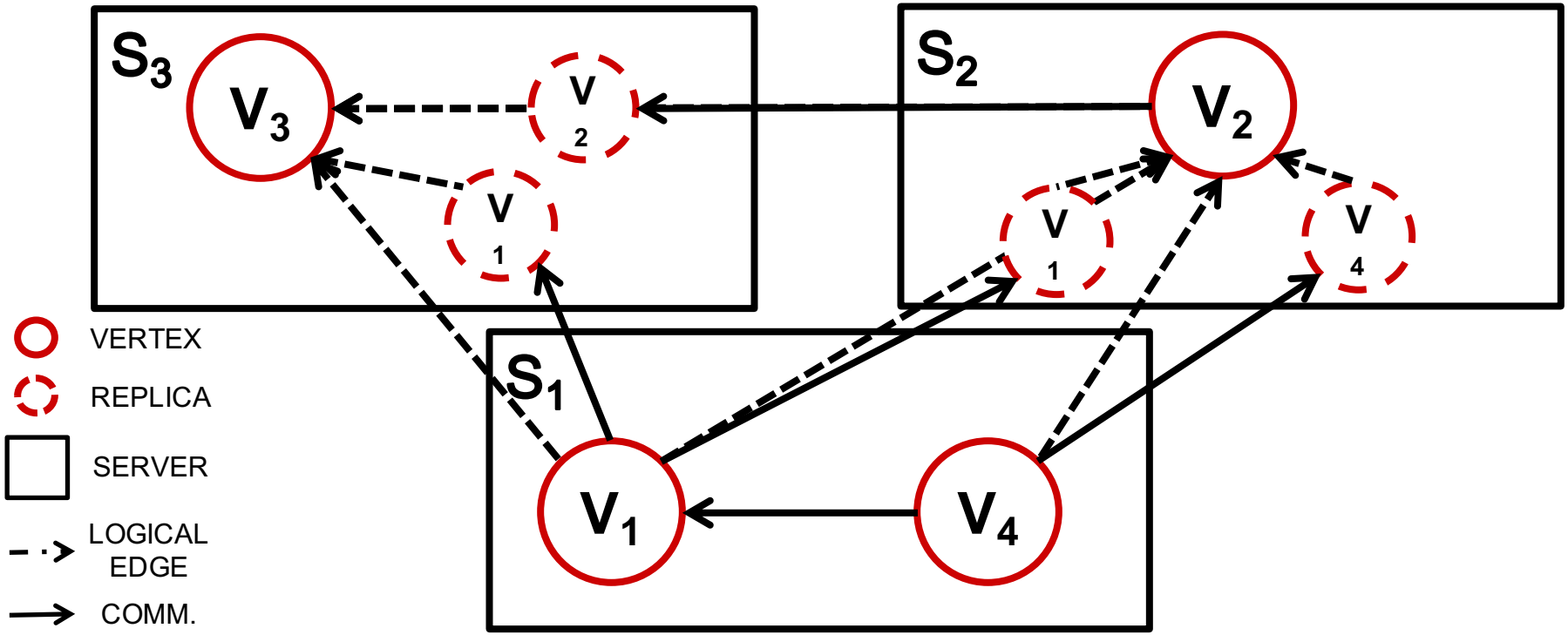
Natural Replication

Example graph



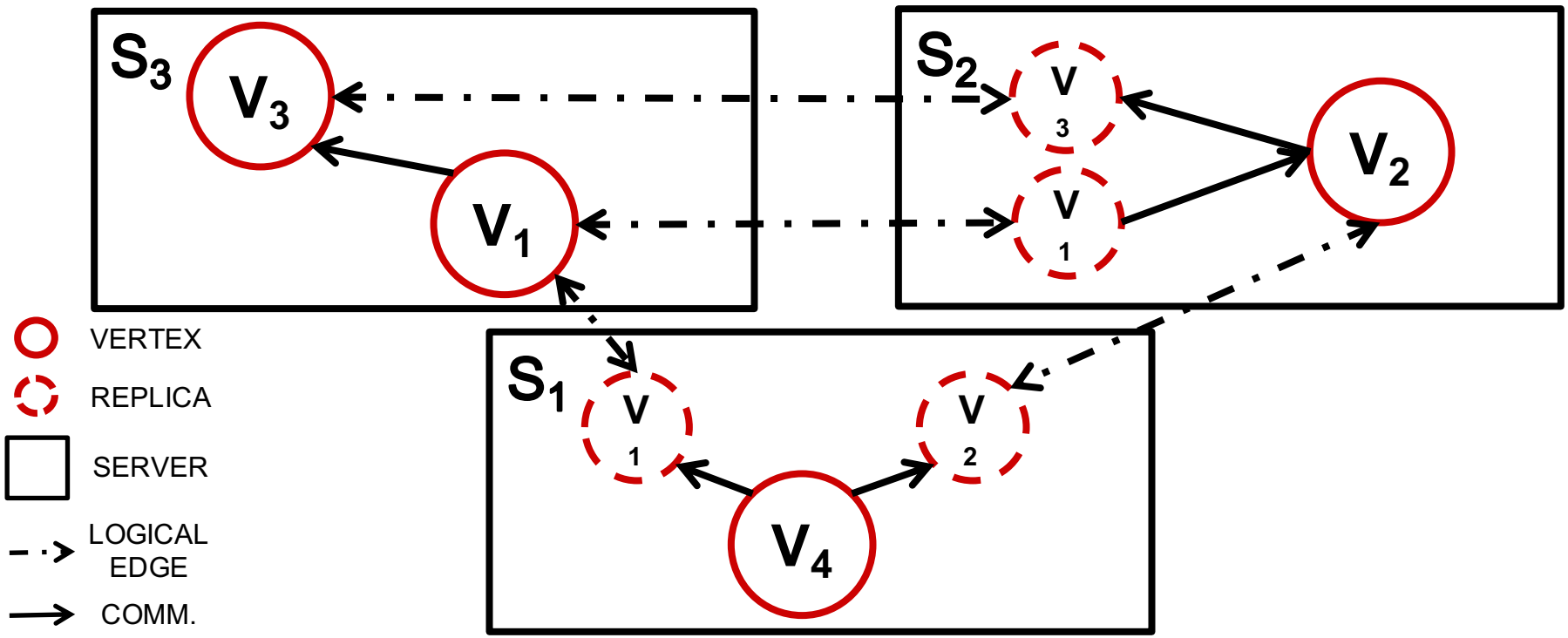
Natural Replication

Out-neighbor replication. Examples: LFGraph, Giraph (old).



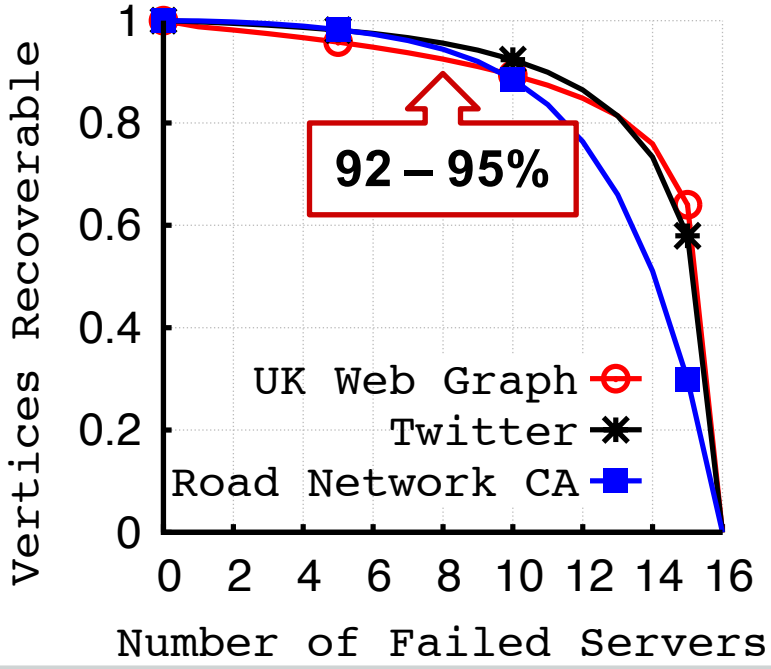
Natural Replication

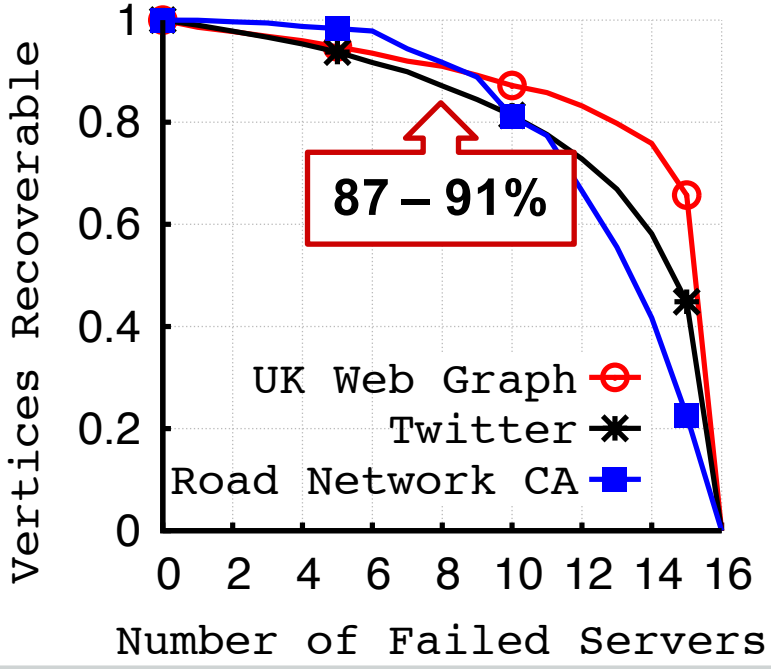
All-neighbor replication. Example: PowerGraph, Giraph (new).



Natural Replication is Robust

If we use natural replication alone, we will incur zero-cost during failure-free execution.





Three R's of ZORR(R)O

Replace

- Membership service for cluster joins and leaves
- Barriers need membership service by design

Rebuild

- Replacements receive state in parallel with initialization
- Rebuild of each server independent of others – solves cascading failures!

Resume

- Computation resumes from the beginning of failure iteration

How does it perform in practice?

Applications:

- PageRank
- Single-source shortest paths (SSSP)
- Connected components (CC)
- K-core decomposition

Tech Report [Pundir2015] additionally evaluates:

- Graph coloring
- Triangle count
- Group-source shortest paths
- Approximate diameter

How does it perform in practice?

Setting: 16 machines (2 x 4 core Intel Xeon processors with hyperthreading – 16 virtual cores, 64 GB RAM, SSDs) inter-connected by 1 Gbps network.

PageRank Inaccuracy Metrics* for k = 100

* Mitliagkas et al. FrogWild!: Fast, PageRank Approximations on Graph Engines, VLDB 2015

Top-k Lost (TL):

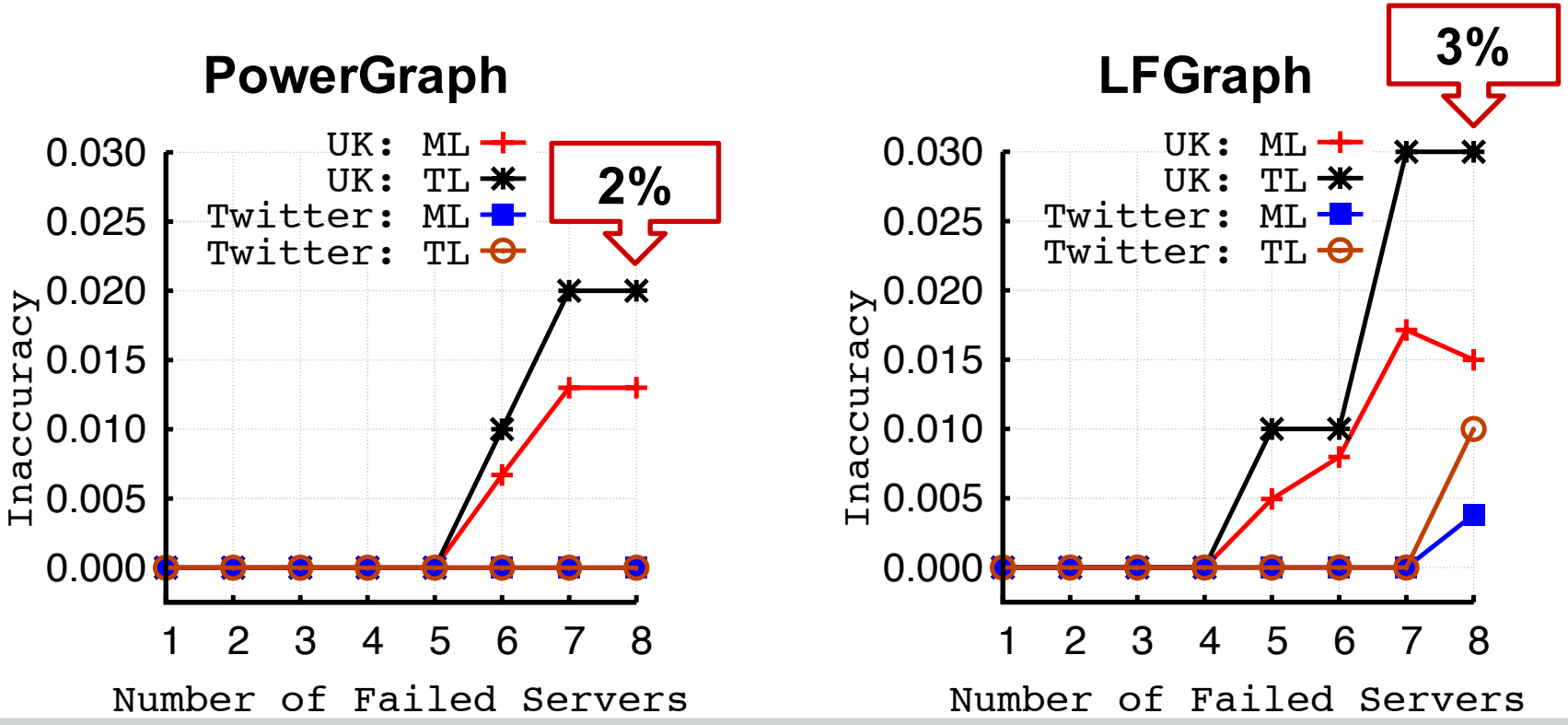
- Fraction of original top-k PageRanked vertices lost.
- How many top PageRank vertices are lost?

Mass Lost (ML):

- Fraction of original top-k PageRank mass/weights lost.
- What is the relative importance of lost vertices in the rankings?

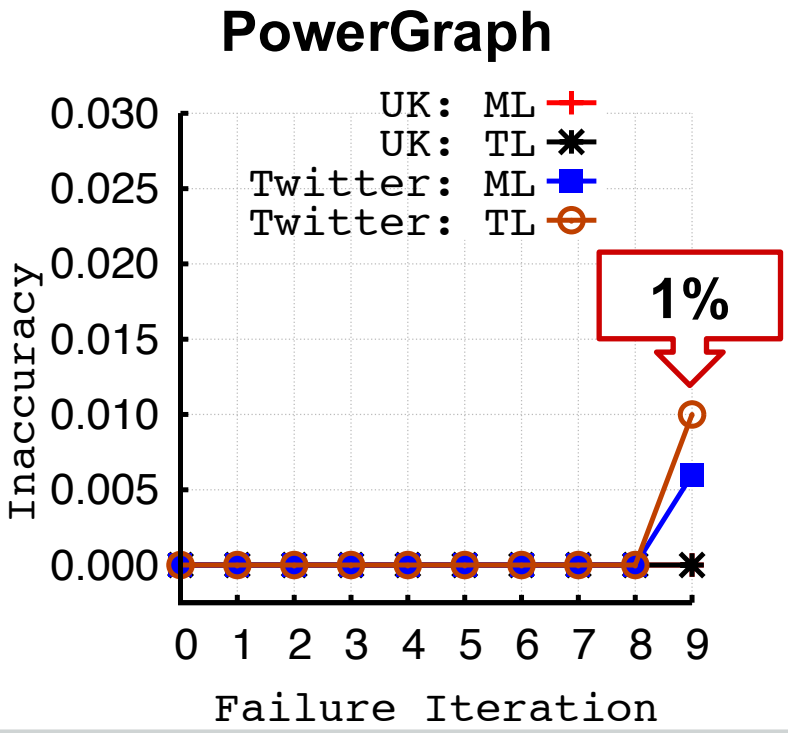
PageRank Evaluation on 16 servers

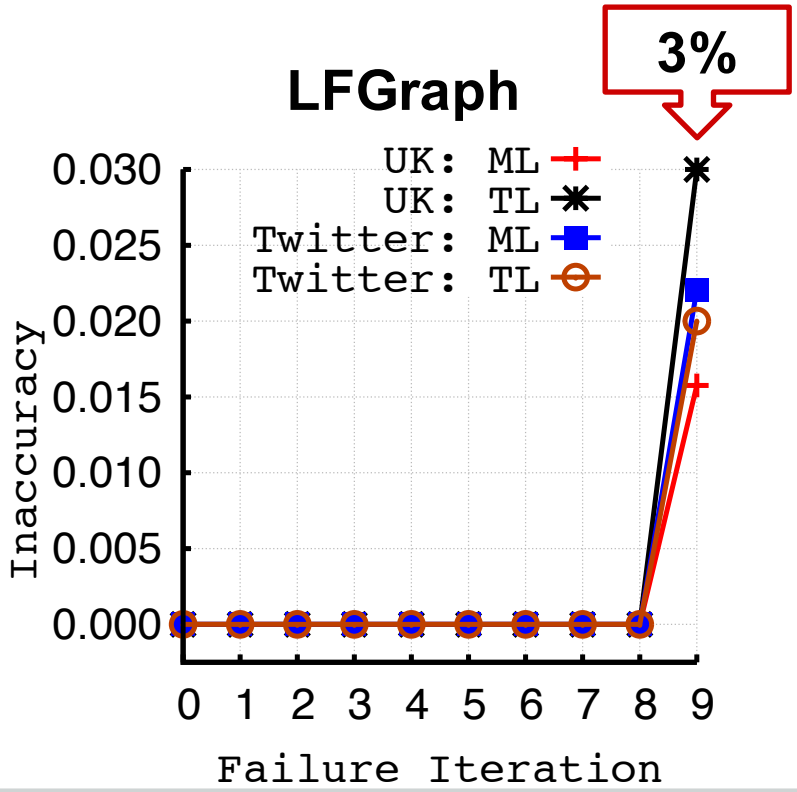
Inaccuracy as a function of the number of failed servers – failures occur in middle iteration (5th iteration)



PageRank Evaluation on 16 servers

Inaccuracy as a function of the failed iteration number – quarter of the servers fail (4 out of 16)





Other Applications

Algorithm	PowerGraph	LFGraph
PageRank	**2 %**	**3 %**
Single-Source Shortest Paths	**0.0025 %**	**0.06 %**
Connected Components	**1.6 %**	**2.15 %**
K-Core	**0.0054%**	**1.4 %**
Graph Coloring*	5.02 %	NA
Group-Source Shortest Paths*	0.84 %	NA
Triangle Count*	0 %	NA
Approximate Diameter*	0 %	NA

*Evaluated in Tech Report [Pundir2015]

Recovery Time

- Zero-cost during common case failure-free execution.
- Recovery time is masked by initialization.
- Additional recovery time is a small fraction of average iteration time and independent of application.

Recovery Network Overhead

- Network overhead during recovery is a fraction of average iteration's network usage.

- If multiple replicas available, only one participates in rebuilding – reduces network consumption by as much as 90% in PowerGraph and balance it across machines.

Effect of Partitioning Strategy

- Compare Random with Grid and Oblivious strategies of PowerGraph.
- Less than 1.2% decrease in accuracy across PageRank, SSSP, CC, K-core applications.

Conclusion

- Checkpointing should be avoided, for distributed graph processing systems, at all costs.
- Distributed graph processing involves natural replication of graph state: 87-95% state recoverable even when half servers fail.
- Utilizing natural replication opportunistically leads to a zero-overhead reactive recovery protocol called Zorro.
- Zorro is accurate, fast, cheap, scalable and resilient.
- We believe Zorro opens up possibility of reactive recovery in other systems.

http://dprg.cs.uiuc.edu/

Backup Slides

SSSP Inaccuracy Metrics

Paths Lost (PL):

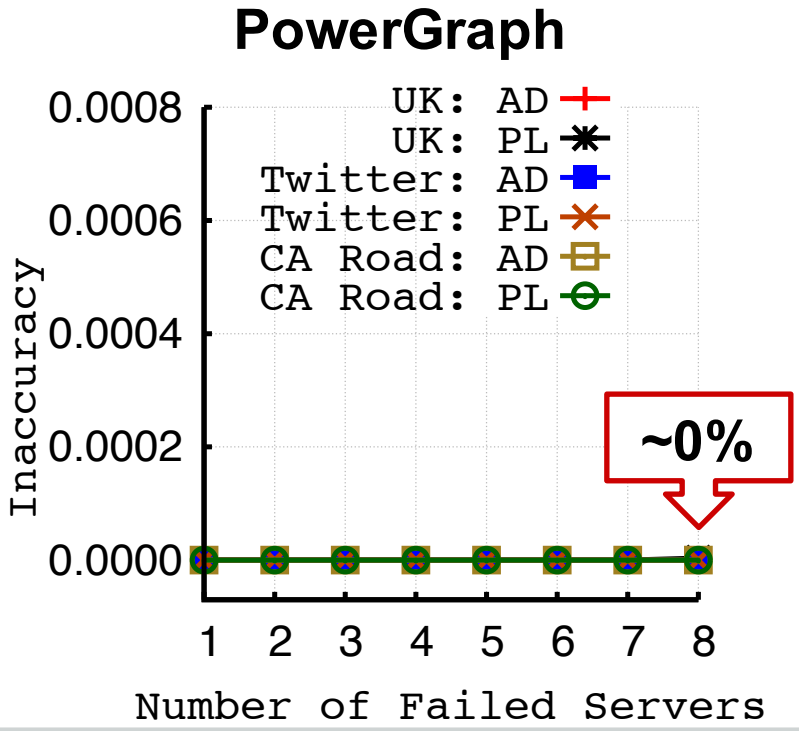
- Fraction of reachable vertices with lost paths after failures.
- How many shortest path values are lost?

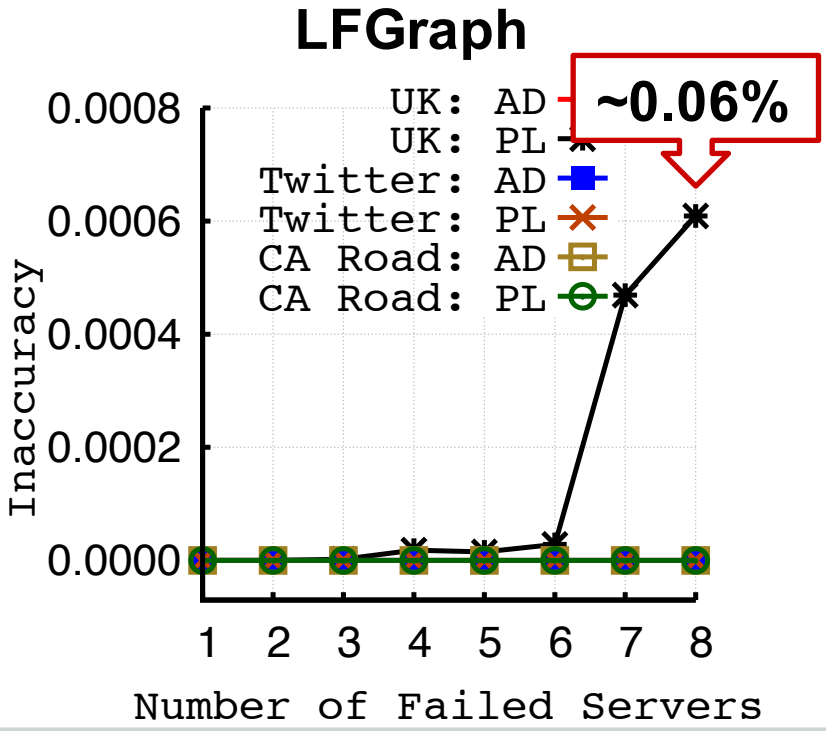
Average Difference (AD):

- Average normalized difference in shortest path values.
- How do the new shortest path values differ from original values?

SSSP Evaluation

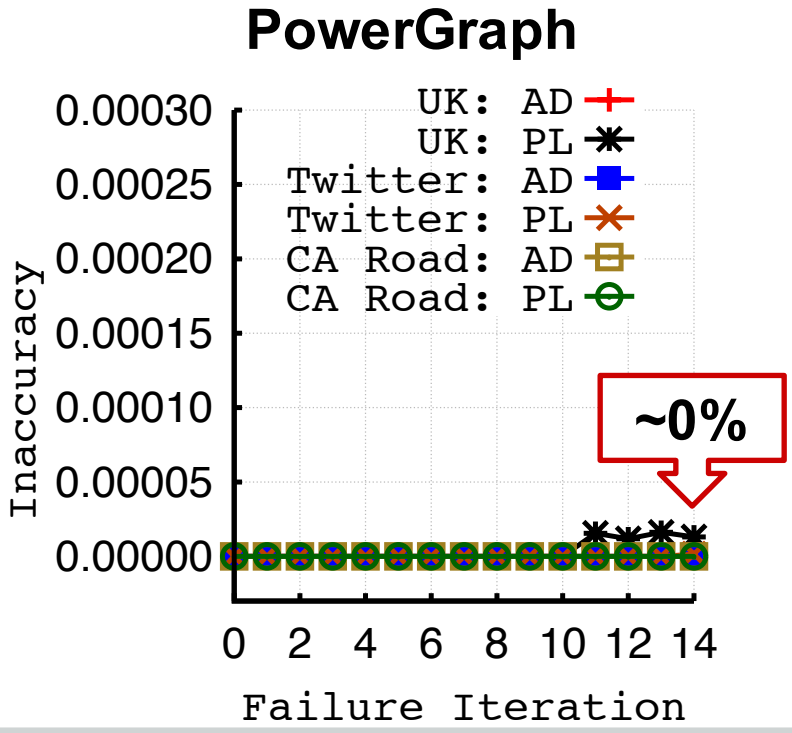
Inaccuracy as a function of the number of failed servers – failures occur in middle iteration (5th iteration)

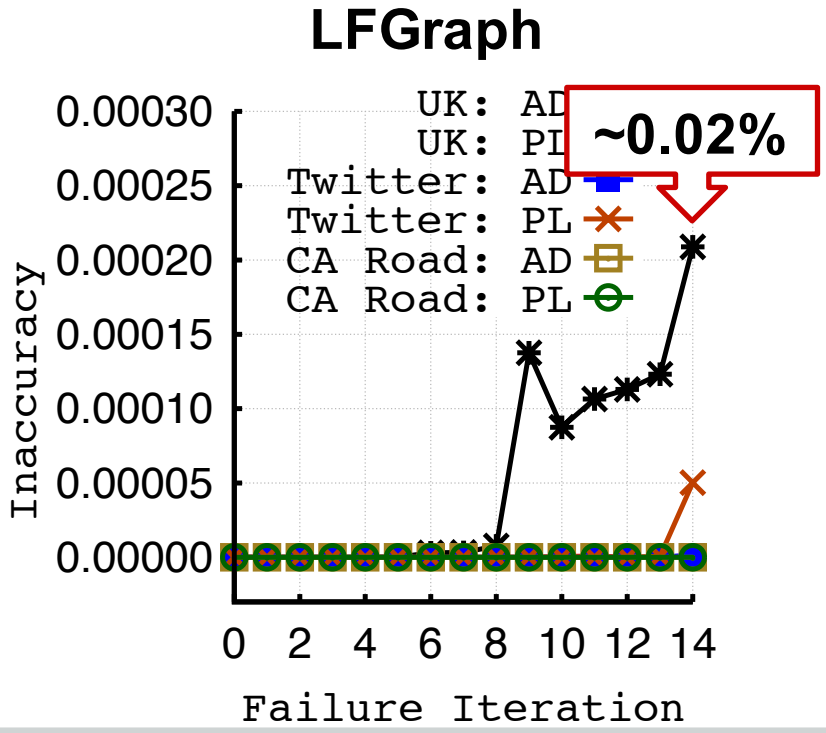




SSP Evaluation

Inaccuracy as a function of the failed iteration number – quarter of the servers fail (4 out of 16)





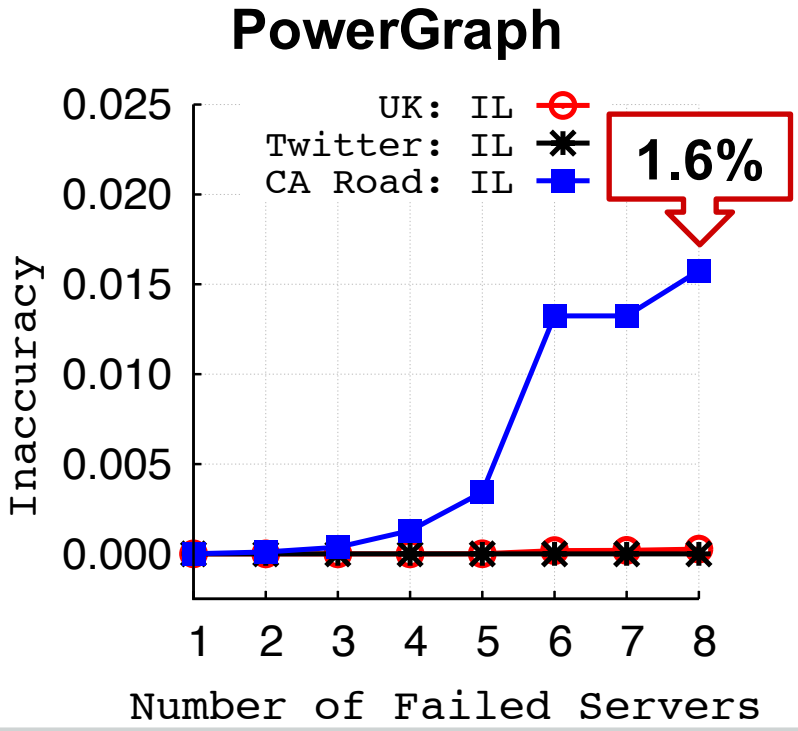
CC Inaccuracy Metric

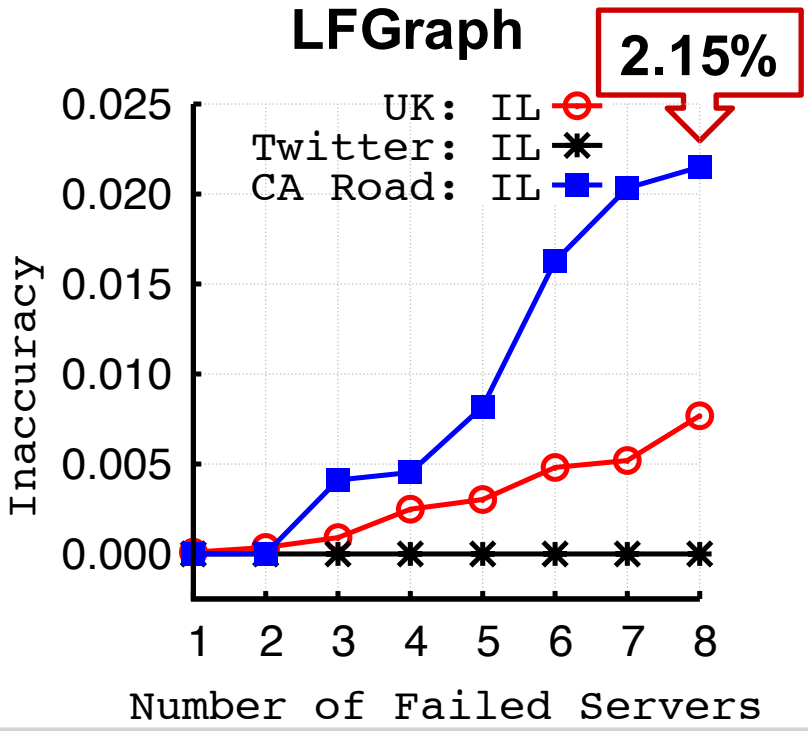
Incorrect Labels (IL):

- Fraction of vertices with a different label i.e., component.
- How many vertices have an incorrect component label?

CC Evaluation

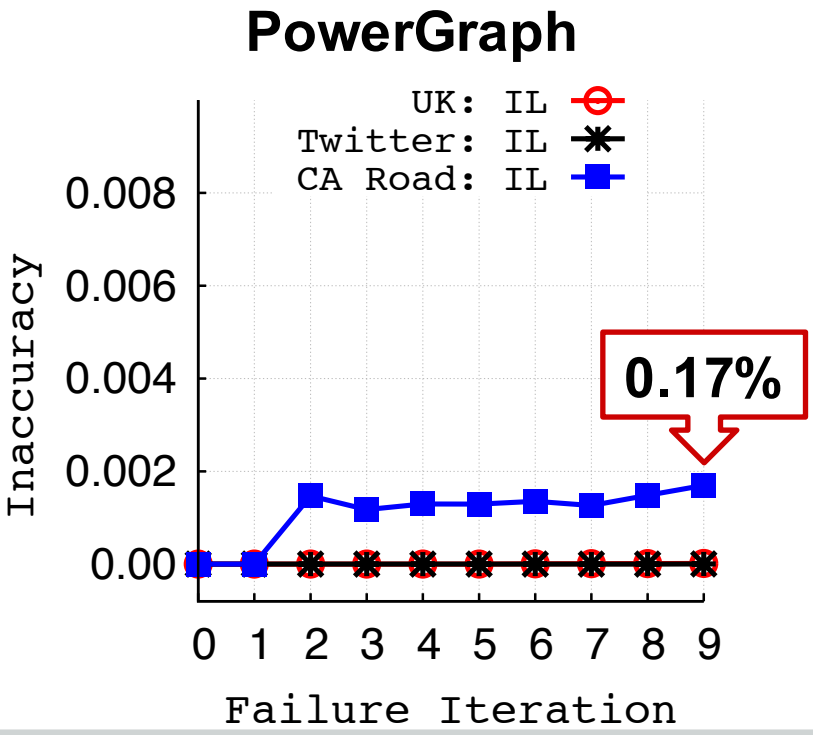
Inaccuracy as a function of the number of failed servers – failures occur in middle iteration (5th iteration)

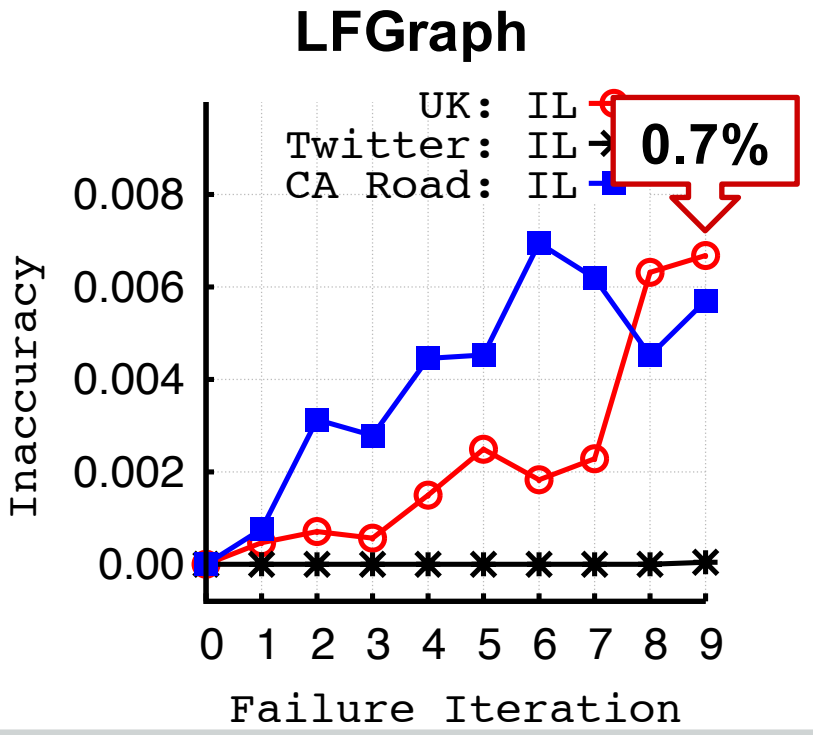




CC Evaluation

Inaccuracy as a function of the failed iteration number – quarter of the servers fail (4 out of 16)





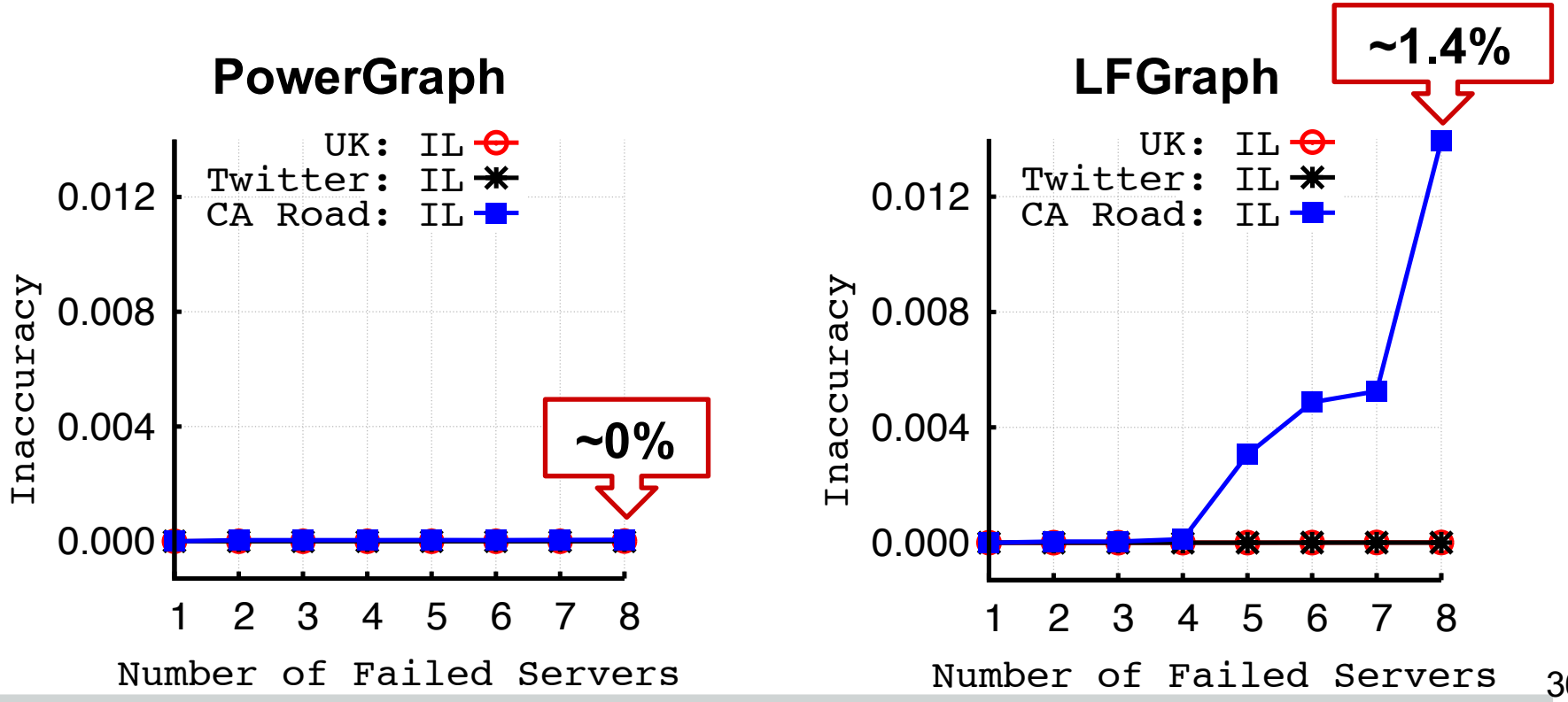
K-Core Inaccuracy Metrics

Incorrect Labels (IL):

- Fraction of vertices with a different label i.e., binary value representing inclusion in induced k-core sub-graph.
- How many vertices have an incorrect label?

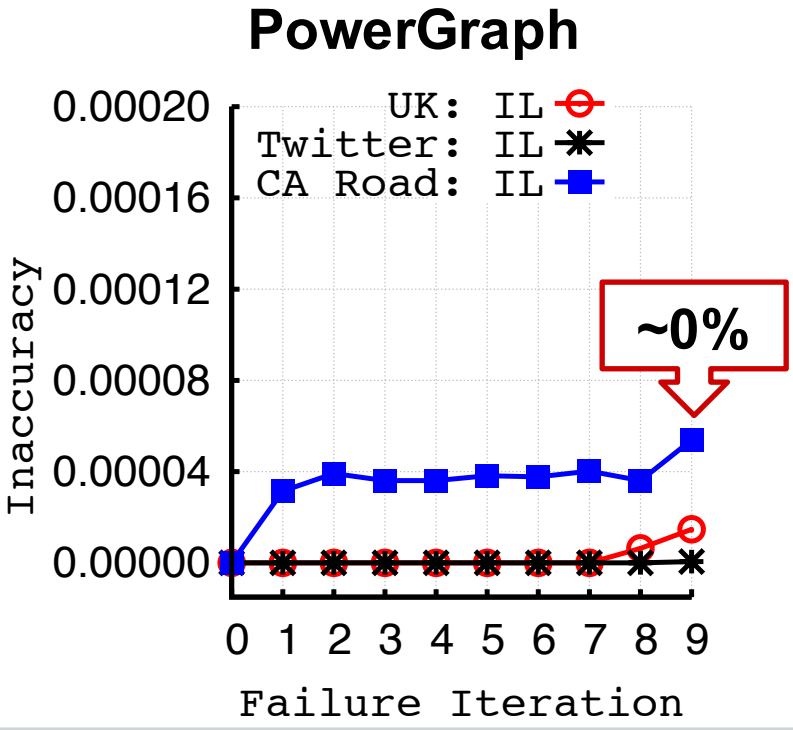
K-Core Evaluation

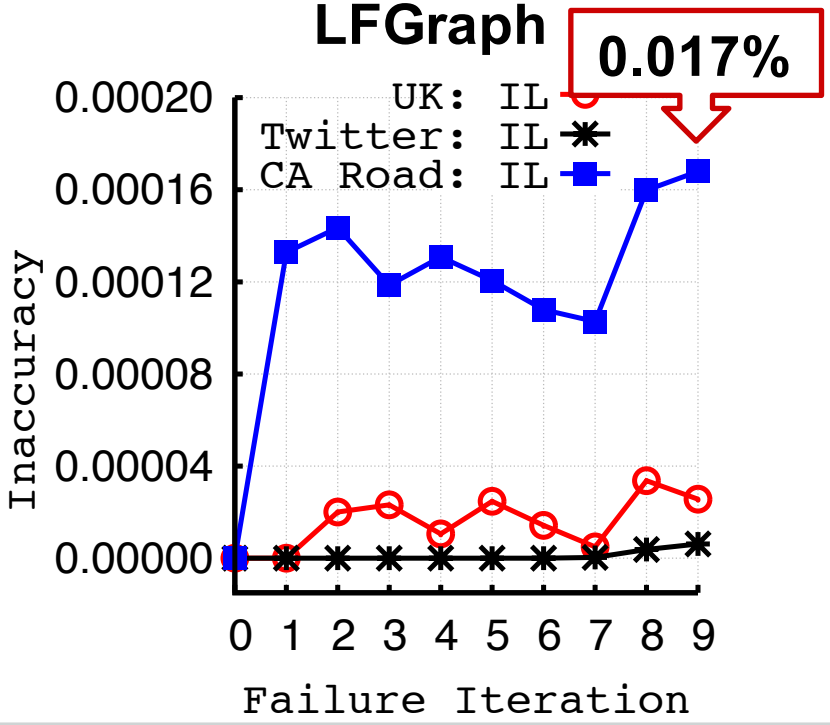
Inaccuracy as a function of the number of failed servers – failures occur in middle iteration (5th iteration)



K-Core Evaluation

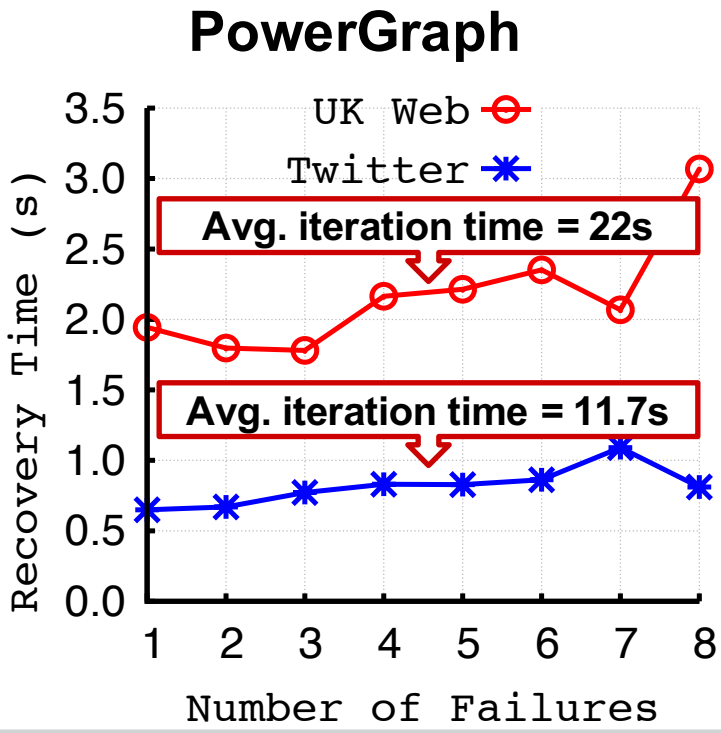
Inaccuracy as a function of the failed iteration number – quarter of the servers fail (4 out of 16)

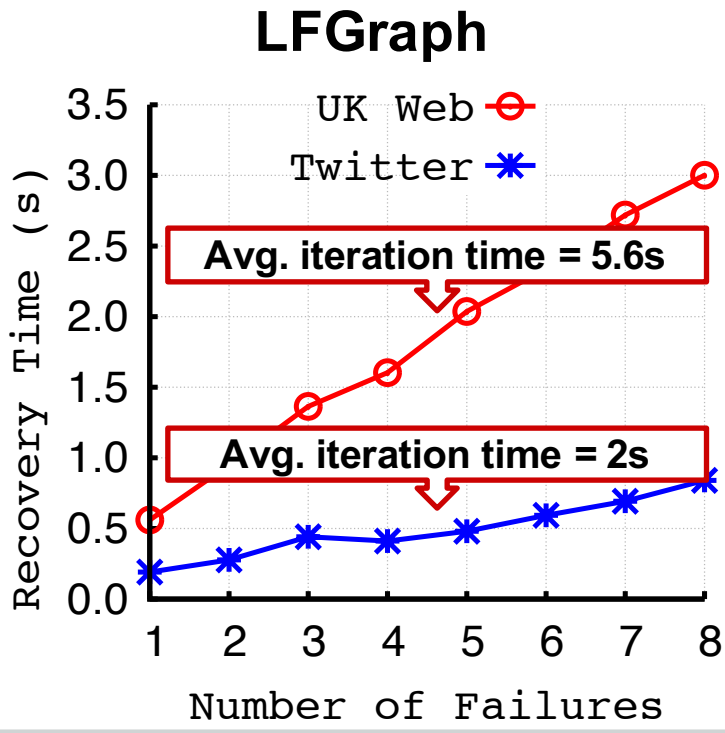




Recovery Time

Recovery time (merging received state) is independent of application

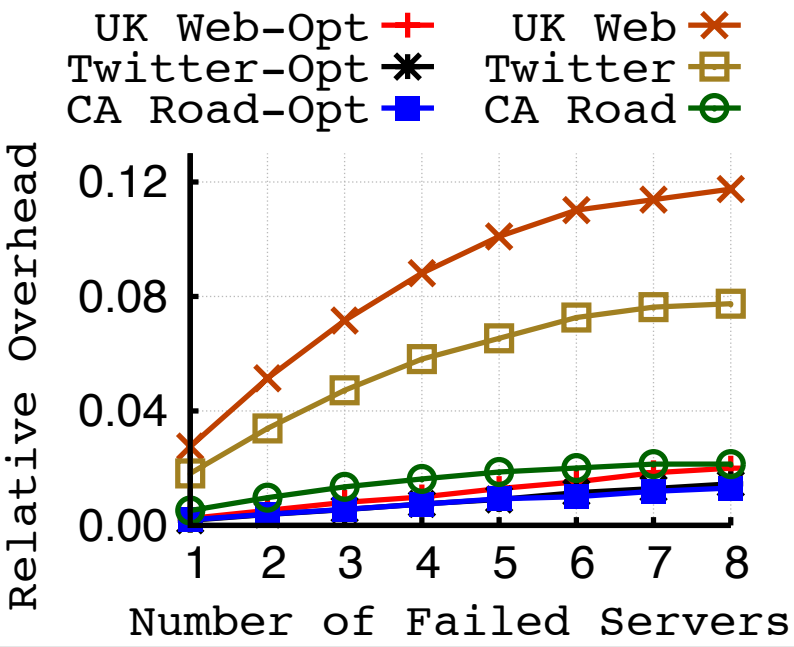




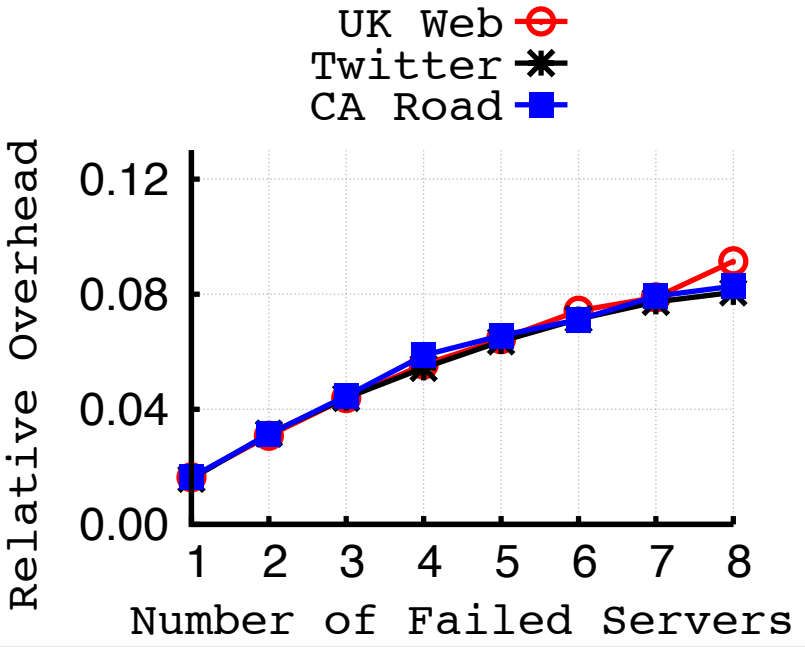
Network Overhead

Network overhead is a fraction of an average iteration's network usage

PowerGraph

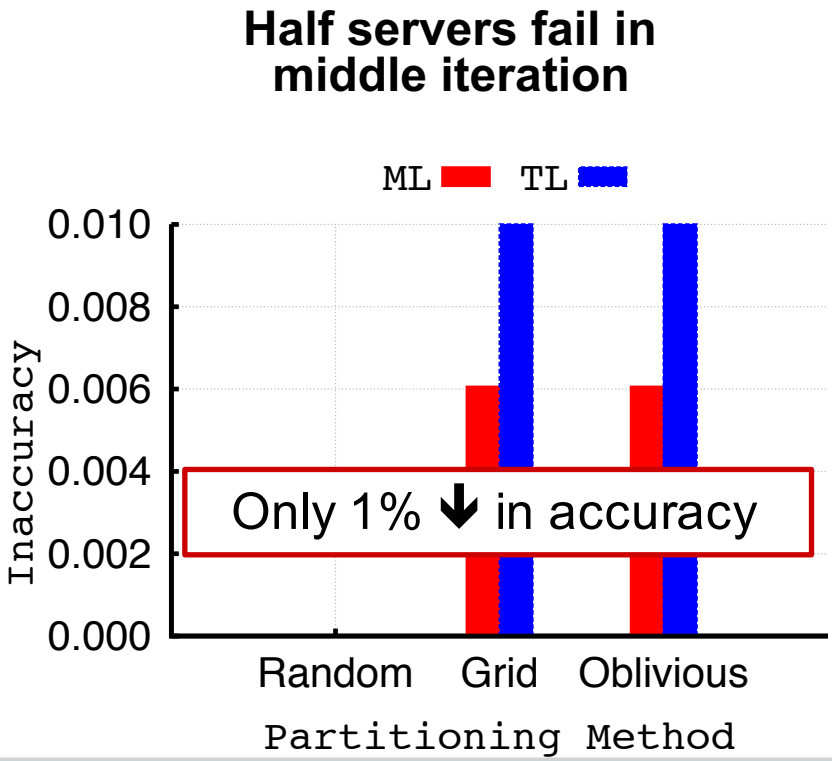


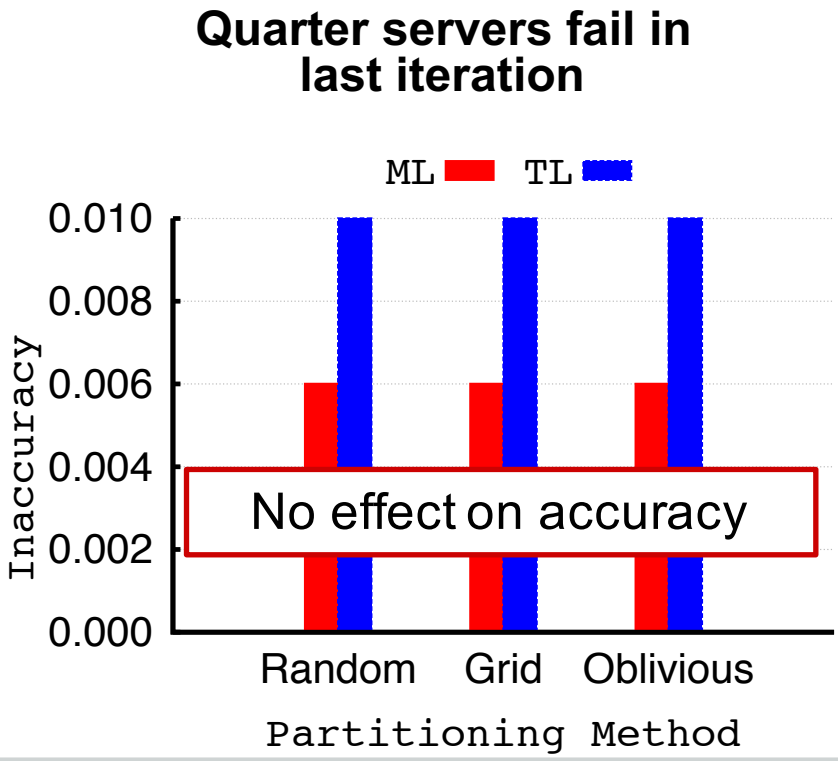
LFGraph



Effect of Partitioning Strategy, decrease or increase or effect

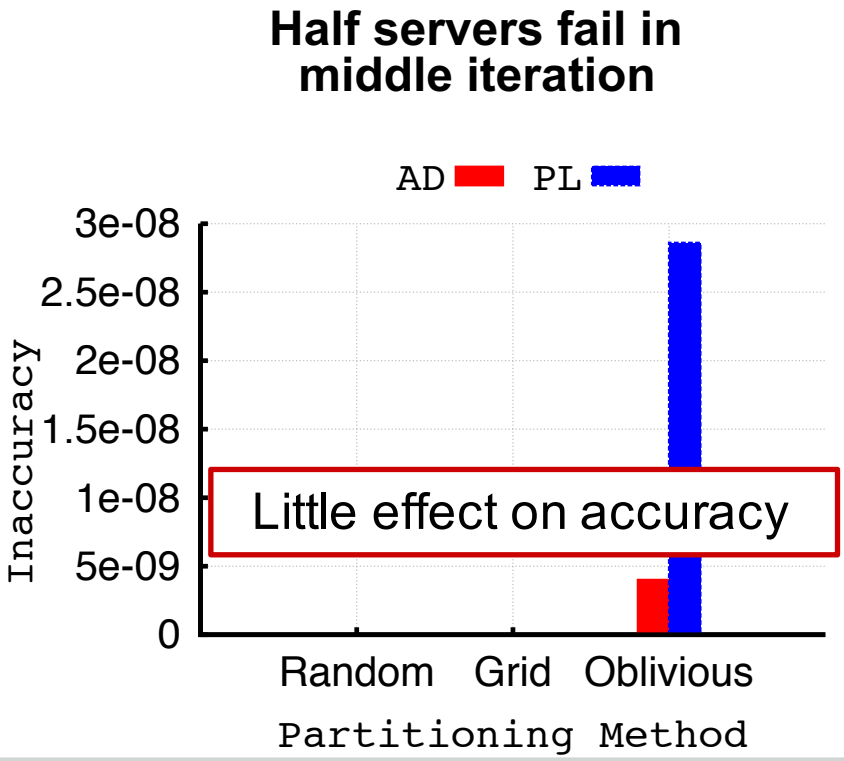
PageRank

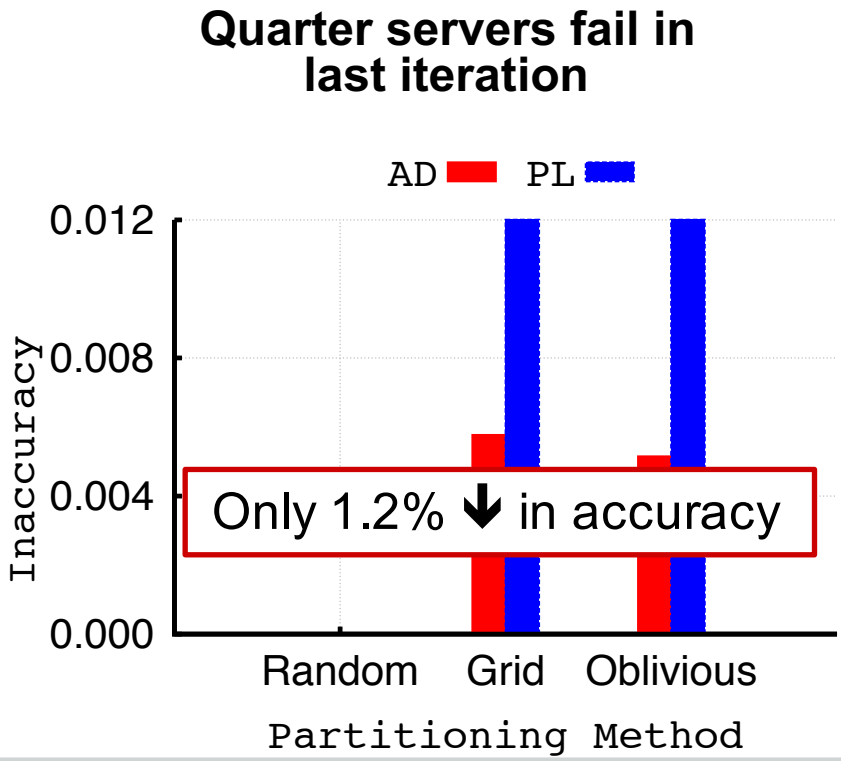




Effect of Partitioning Strategy

SSSP





References

[PrestaFB2014]: Presta et al. Large Scale Graph Partitioning with Apache Giraph. Facebook Engineering Blog, 2014. https://code.facebook.com/posts/274771932683700/large-scale-graph-partitioning-with-apache-giraph/

[FBQ22015]: Facebook Q2 Reports. http://investor.fb.com/releasedetail.cfm?ReleaseID=924562

[TwitterStats2015]: Twitter Company Statistics. https://about.twitter.com/company

[Myers2014]: Myers et al. Information Network or Social Network?: The Structure of the Twitter Follow Graph. WWW Companion 2014.

[GoogleSearch2015]: Google Inside Search – How Search Works. http://www.google.com/insidesearch/howsearchworks/thestory/

[Pundir2015]: Pundir et al. Zero-Cost Reactive Failure Recovery in Distributed Graph Processing. IDEALS Technical Report, 2015. https://www.ideals.illinois.edu/handle/2142/75959

References

[Yigitbasi2010]: Yigitbasi et al. Analysis and Modeling of Time-related Failures in Large-scale Distributed Systems. GRID 2010.