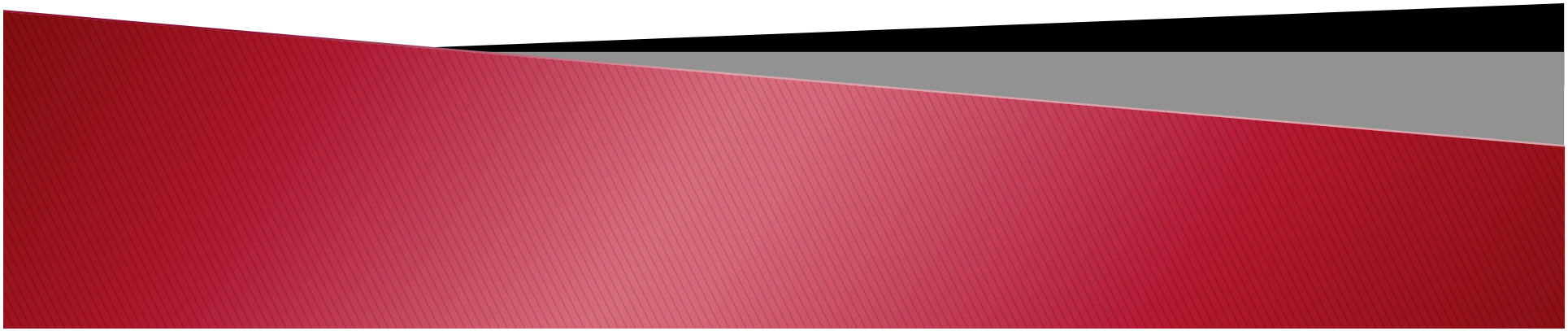
UNM SCHOOL of ENGINEERING
Department of Computer Science

A Gossip-Based Approach to Exascale System Services

Philip Soltero, ***Patrick Bridges***, and Dorian Arnold
University of New Mexico

Mike Lang
Los Alamos National Laborotory



Building Exascale System Services

- Need large-scale services at huge node counts
  - Job launch, power monitoring/control, load balancing, etc.
  - System-wide communication a major challenge here
- Have to worry about all of address the standard exascale and distributed system design concerns
  - Power, Resilience
  - Scalability, Consistency
- We've traditionally designed HPC system services like they were HPC applications:

Synchronous, Structured, and Global

How much do we need consistency?

- Same tired old idea: Discard consistency for scalability or resilience
- For what services does this make sense?
  - Dependent on hardware and programming model
  - Past work in this direction has for load balancing, other services
- What kind of weakly consistent communication to use?

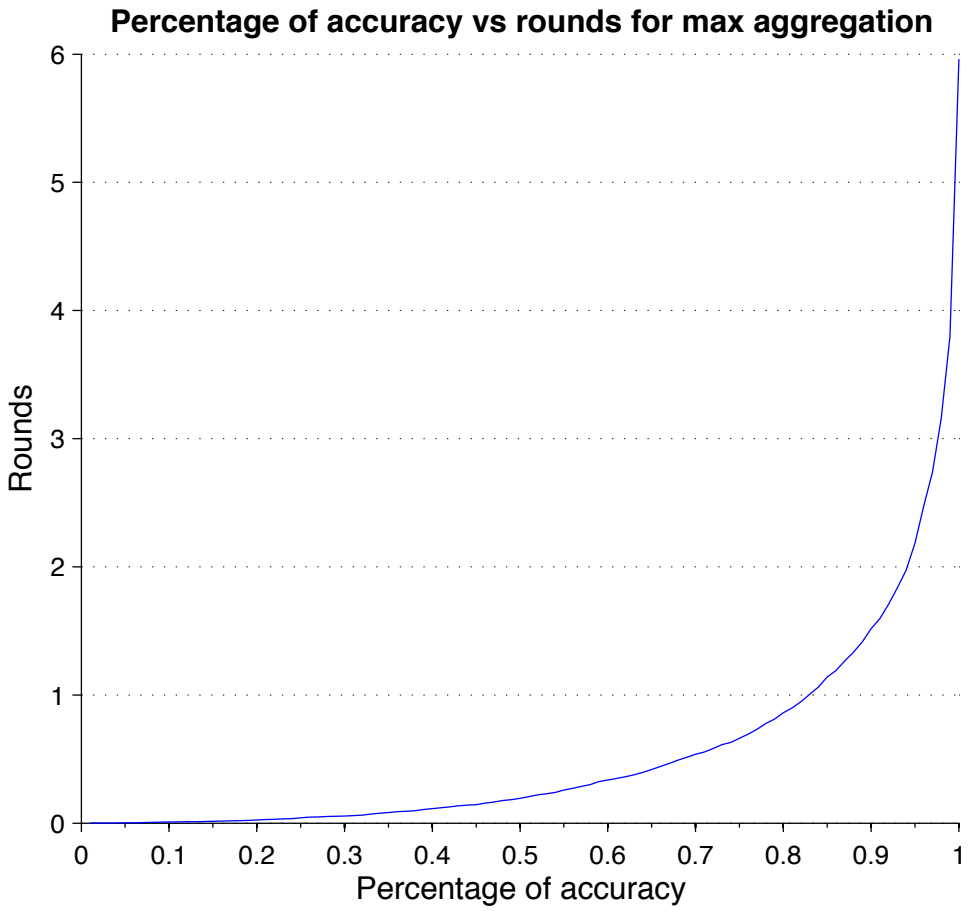
Gossip-based Communication

- Popular recent distributed system technique
  - Round-based protocol
  - Each round: Every node exchanges information with small ***random*** set of nodes
  - Information propagates ***epidemically*** throughout system
  - Design so global data view converges to correct value
- Robust to failures; no global communication coupling

Some things are hard to Gossip

Some types of aggregation are easier than others

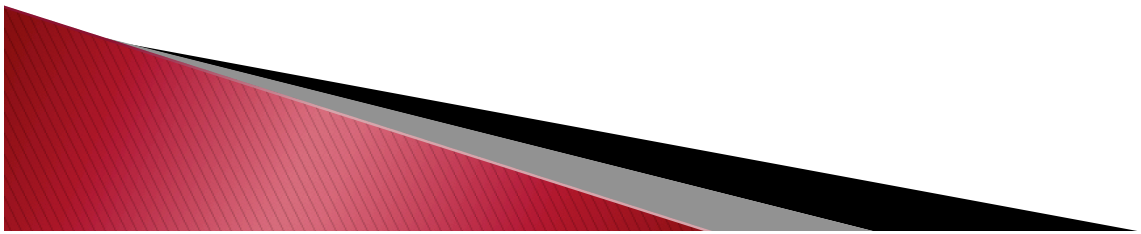
- Idempotent operations (max, min, etc.) easy to do
- Average, Sum, etc. are more difficult – simple pairwise exchanges are insufficient
- Can use more complex protocols for computing global sums



Gossip can also be slow

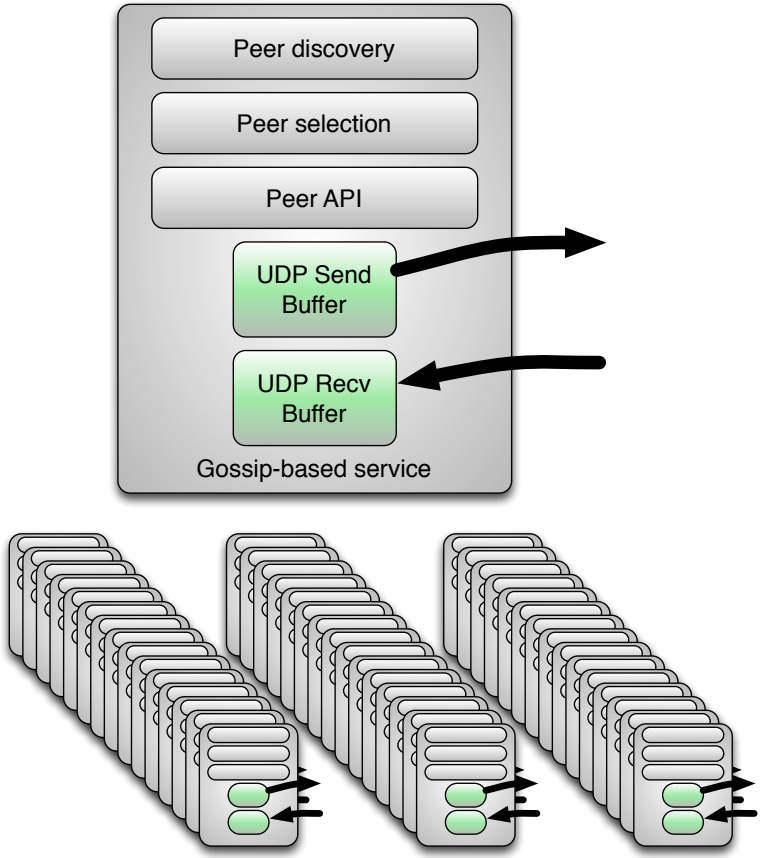
- Takes some number of synchronized rounds for results to converge towards true value
- Different nodes have different values at different times
- When to use the current value or start a new round?
- Few well-understood roundless gossip protocols

Can we actually build useful exascale services with this?



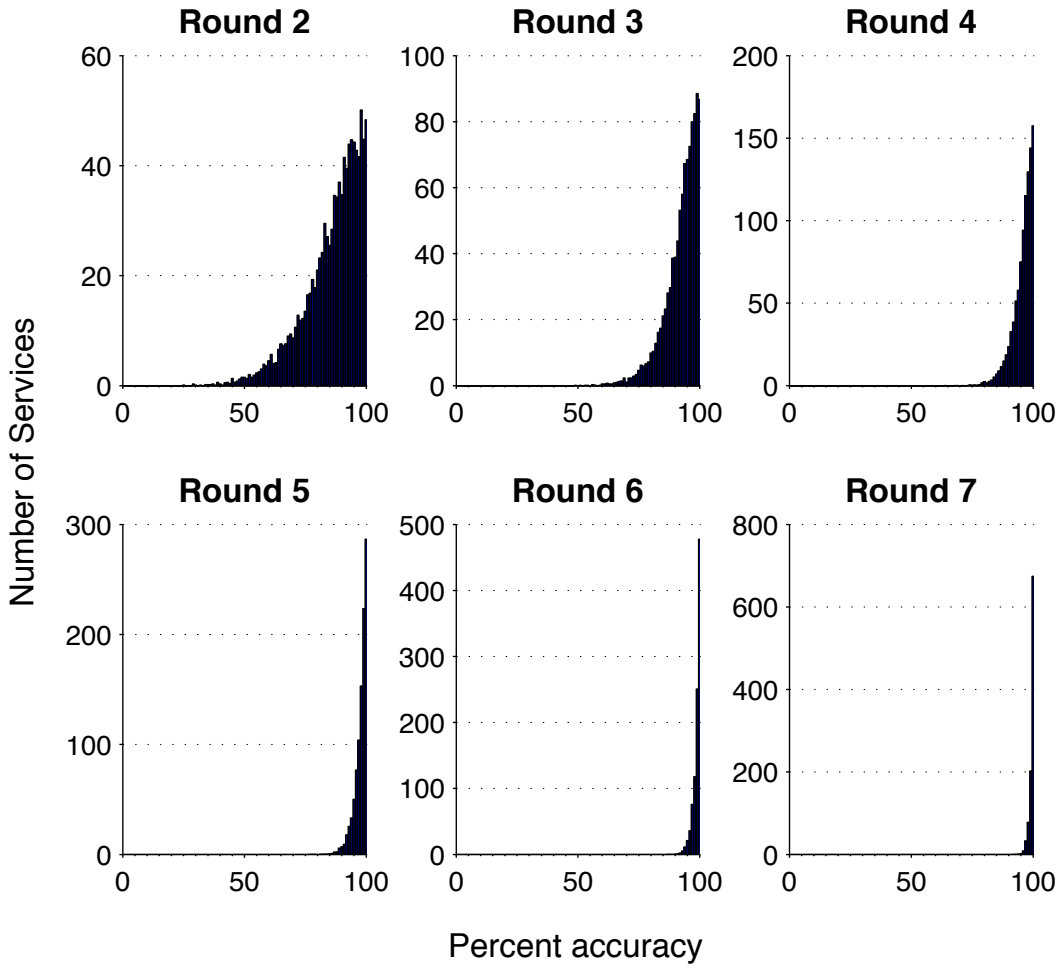
Implementation

- Built a simple UDP gossip library for testing gossip algorithms
- Currently uses a ***fixed*** view of potential peers to select from



How well does Gossip converge?

- 1000 participants computing averages
- How close is each node to the real average after each round?
- Very high accuracy in about 6-7 rounds



Prototype: Gossip-based Power Control

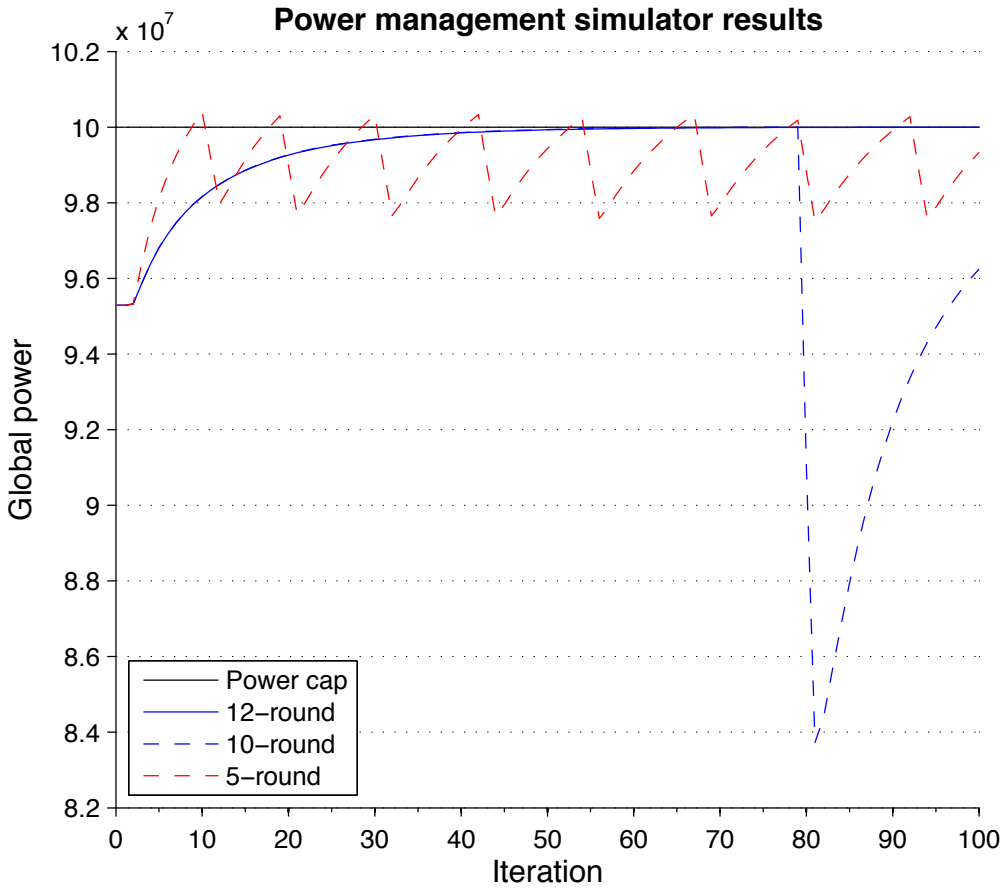
- Goal: Simple power control scheme to examine the limitations of gossip for exascale system service design
- Given:
  1. Cap on average local power consumption (global cap)
  2. Multiple available power gears (F/V pairs)
  3. Local power measurements
- Approach:
  1. Use gossip to estimate global power usage
  2. Locally change gears to help converge global average towards desired value

Evaluation

- Simulate effects of power consumption control
- Process:
  1. Each node sets local power
  2. Simulator determines resulting global power usage
  3. Nodes are given communicated global power usage based on (scaled) accuracy profiles
  4. And on around the loop
- Assume perfectly balanced load
- 5 energy gears (1200MHz/1.2V to 2000MHz/2.0 V)

Impact of Changing Accuracy on Power Management Decisions

- Graph is for 1000 participants
- With "enough" rounds we can get sufficient accuracy and hence control
- "Enough" is 24 rounds at exascale



Analysis and Conclusions

- Upsides
  - Can still get reasonable control (in this one case) even when we've discarded any guarantee of complete consistency
  - Gossip is robust to failure (5% failure with a simple failure model didn't impact gossiped value or accuracy)
- Downsides
  - 20-24 rounds (with one peer per round) is non-trivial, corresponds to 10-12 level binary reduction tree (e.g. TBON)
  - Behavior can be poor if accuracy is insufficient
  - Need some fallback to enforce hard limits

Related and Future Work

- Related Work
  - Structured Communication Networks (TBON, CIFTS, etc.)
  - Asynchronous/non-blocking collectives
  - A whole raft of traditional distributed systems studies
- Future Work
  - More thorough resilience studies
  - Experimental study of asynchrony/consistency tradeoffs
  - Feasibility for other exascale services (resilience, etc.)

Acknowledgements

- Faculty and students in the UNM Scalable Systems Lab
- Funding from the New Mexico Consortium and Los Alamos National Laboratory