Investigation of Techniques to Model and Reduce Latencies in Partial Quorum Systems
Ning Tan, Wesley Chow
Electrical Engineering and Computer Sciences, University of California at Berkeley

Introduction

- Distributed Storage System is widely used
  - Google’s BigTable
  - Amazon’s Dynamo
- Efficiency is the key
  - 100ms extra latency in Dynamo $\rightarrow$ 1% decrease in sales
  - 500ms extra latency in BigTable $\rightarrow$ 20% decrease in searches

Distributed NoSQL Data Store

- Parameters $N, R, W$
  - $N$: number of replicas
  - $R$: number of read acknowledgement
  - $W$: number of write returning values
- PBS assumption: the latency distribution across replicas are i.i.d

Fast Retry (Duplicate Replies) Algorithm

- Coordinator sends duplicate requests if no response in 5, 10, 15, 20ms
  - Known to reduce latency in BigTable
- Replicas send a duplicate response with probability $p$
- Trade-offs
  - Sending redundant requests adds to network (and potentially disk) loads in search of lower latencies
  - Systems under high load are typically what will generate long tail latency numbers, and adding to that past a certain point is not beneficial

Graphical Model

- Powerful formalism that brings together probability and graph theory
- Proven to be very useful in bioinformatics, speech/image processing, control theory
- Use Graphical Model to model/predict network traffic
  - Naturally incorporates network topology and time
  - Models correlations between nodes

Learning/Prediction via Graphical Model

- Maximum Likelihood Estimation
  - existence of latent variables (server failure, packets dropping...)
- EM algorithm
  - deals with latent variables
  - convergence has been shown
- Use Junction Tree Algorithm to do exact inference

Implementation and Evaluation

- Modify Apache Cassandra
  - Cassandra sends one data request to the closest node as determined by latency, then sends digest requests to other replicas
  - Our implementation modifies the ReadCallback to send another request if a reply is not received within some time frame
  - Duplicate replies: responding node sends another reply immediately
  - Measured latencies with Cassandra stress utility, reading 1 million keys from a cluster with 4 replicas per key and a consistency level of “quorum”
  - Pull data from LatencyTracker for read statistics
- Tests were run using 8 nodes from the Psi Millennium Cluster
- Apply Graphical Model and PBS to predict the performance of Fast Retry

Results

Conclusions and Future Works

- More accurate modeling of the system
- Fast-retry reduces latencies, but not satisfactory
  - improve result by acquiring more detailed data at long tail
  - identify the most promising techniques for systematically reducing latencies
  - third request at higher percentiles?

Acknowledgment

We would like to thank Peter Bailis, Aaron Davidson, Anthony D. Joseph, John D. Kubiatowicz, Jon Kuroda, Aviad Rubinstein, Shivaram Venkataraman, AMP Lab, RAD Lab and Department of EECS for the help we received throughout the project.

Created with LaTeXbeamerposter: http://www-i6.informatik.rwth-aachen.de/~dreuw/latexbeamerposter.php