ESTIMATING OPERATING CONDITIONS IN A PEER-TO-PEER SESSION INITIATION PROTOCOL OVERLAY NETWORK

JOUNI MAENPAA
GONZALO CAMARILLO

HOT-P2P 2010
ATLANTA, APRIL 23RD , 2010

ERICSSON

ESTIMATION OF OPERATING CONDITIONS

- A decentralized person-to-person communication system being standardized in the Internet Engineering Task Force (IETF)
  - Uses the Chord Distributed Hash Table (DHT)
- Why to estimate operating conditions?
  - Adaptive behavior
    - DHT-based overlays are difficult to configure
      - The current practice is to configure them statically
    - Adaptive behavior is needed to make them scalable
      - But requires knowledge about the current state of the system
  - Performance monitoring
    - What is the size of the overlay?
    - What is the churn rate?
- The paper
  - Evaluates the accuracy of existing size and churn rate estimation algorithms
  - Studies how they can be improved to achieve higher accuracy

P2PSIP SIMULATOR

- Used in the simulations carried out for the paper
- An event-driven, message-level simulator
- Implemented in Java
- The code base is the same as in our real-world P2PSIP implementation used in PlanetLab and in previous work
- Uses
  - Peer-to-Peer Protocol (P2PP) as the peer protocol
  - Chord DHT
- Topology generator
  - Assigns peers randomly to 206 locations around the world
  - The locations correspond to PlanetLab sites
  - Pairwise delays between peers were set based on real pairwise delays measured between PlanetLab sites

EXPERIMENTS

- Lookup traffic consisted of lookups related to Voice over IP (VoIP) calls, instant messaging, and presence
  - Call rates modeled according to busy hour call volumes
  - Arrival of calls modeled as a Poisson process
  - The number of buddies a user has follows the power law distribution with an average of 22
- Arrival and departure of peers modeled as a Poisson process
  - 10% of departures were crashes
- Chord parameters
  - Chord stabilization interval was set to 15s
  - Size of Chord routing table was 35 peers
  - Size of failure history was set to 25% of routing table size (i.e., 9)
  - Keepalive interval was 30s
- Duration of the simulated period was 4.5h
- Maximum network size was 10049
  - Maximum inter-arrival time: 500ms
  - Maximum inter-departure time: 500ms

BASIC ESTIMATION MECHANISMS

- Network size *N*
  - Use the density of peer-IDs in Chord's neighbor table
  - *d*: the average inter-peer distance
  - *m*: the number of bits in Peer-IDs

$$N = \frac{2^m}{d}$$

- Join rate λ
  - Use the uptime of the peers in the routing table
  - *Ages*: an array containing the ages of peers in increasing order
  - *r*: the size of the routing table

$$\lambda = \frac{N}{4} \times \frac{1}{Ages[r/4]}$$

- Leave rate *μ*
  - To calculate the estimate, measure the time *Tk* it takes to detect *k* failures
  - *M*: the number of unique peer-IDs in routing table
  - *Tk*: time between first and last failure

$$\mu = \frac{k}{M \times T_k}$$

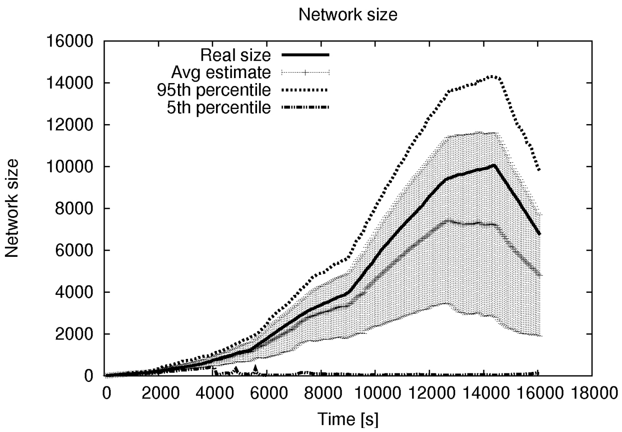
OPTIMIZATIONS

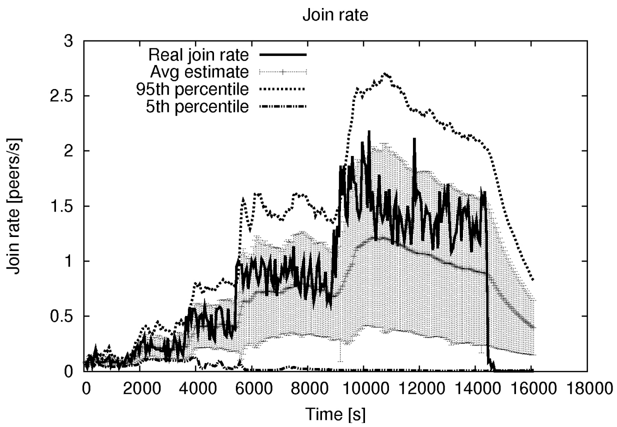
Configuration	Optimizations
Basic	No optimizations, basic estimation techniques used
Leave rate optimizations	›Extended failure history (with identities and network sizes) ›Joining peers download the failure history ›Modified leave rate calculation
Two network size estimates	Calculate a secondary network size estimate using the average distance of finger pointers from the ideal position
Estimate sharing	Peers share their estimates by piggybacking them in P2PP messages
All optimizations	›Percentiles instead of weighted averages when processing shared estimates ›Improved failure detection ›Modified calculation of initial leave rate estimate ›Old entries dropped from failure history

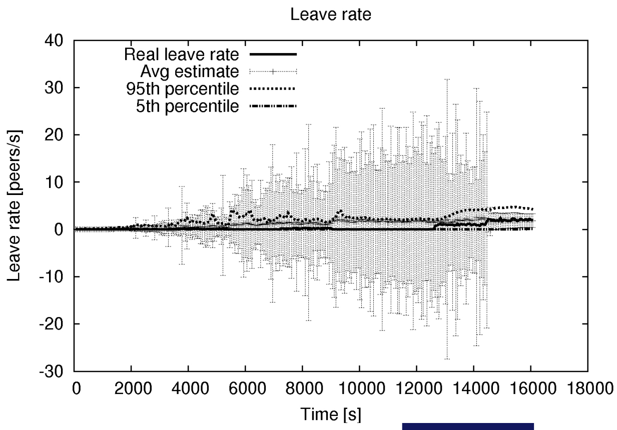
RESULTS – NO OPTIMIZATIONS

- On the average, the size estimate is wrong by 36.6%
- The join rate estimate is wrong by 54.4%
- The leave rate estimate is wrong by 293%
  - The high standard deviation in the figure is caused by poor estimates of newly joined peers
- The error bars in the figures represent the standard deviation

$$avg\,error = \frac{\sum_{i=0}^{n} \left|real - est_i\right|}{n}$$

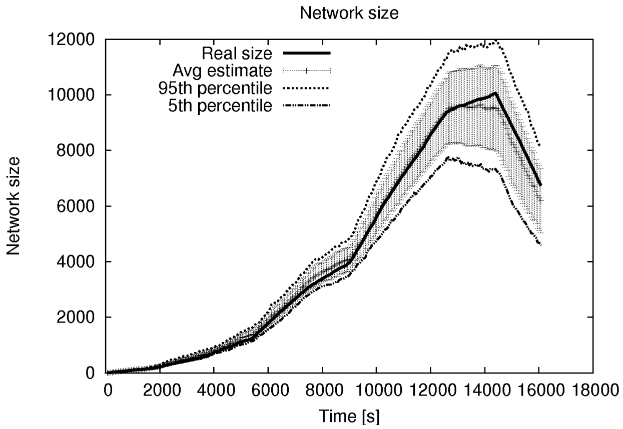


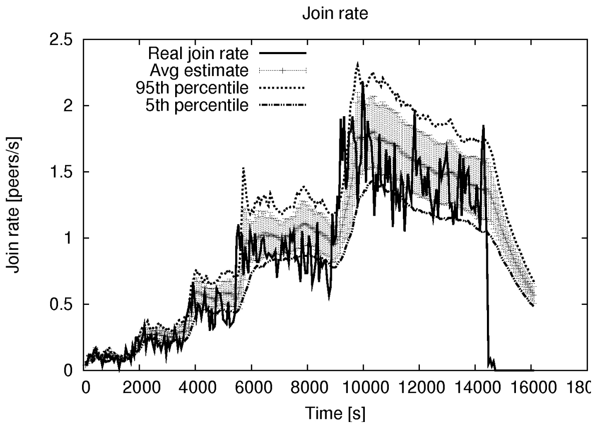


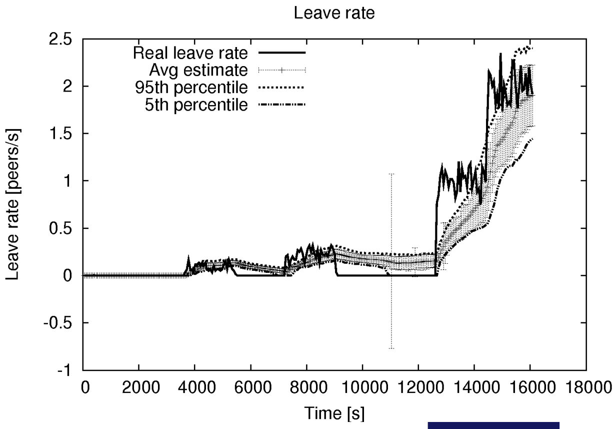


RESULTS – ALL OPTIMIZATIONS

- Network size estimate wrong by only 10.6%
- Join rate estimate wrong by 35.2%
  - Average join rate estimate no longer underestimates the real join rate
  - Also sudden changes detected faster
  - The highest error when the real join rate drops to zero
- Leave rate estimate wrong by 47.6%
  - Standard deviation dramatically lower
  - Still reacts quite slowly to sudden increases in leave rate







SUMMARY

- In the paper, we show how to improve the accuracy of mechanisms estimating the operating conditions of a running P2PSIP overlay
- The accuracy of existing mechanism is not satisfactory
- Several optimizations were studied
- Only passive mechanisms were used
- In practice, peers need to share their estimates by piggybacking them to overlay stabilization messages
- When all optimizations are used, the improvement is
  - 239% for network size
  - 55% for join rate
  - 515% for leave rate estimate

ERICSSON