CollectCast: A Tomography-Based Network Service for P2P Streaming

Mohamed Hefeeda, Ahsan Habib, Dongyan Xu, Bharat Bhargava
Computer Sciences Department, Purdue University

1. Motivation
- Target environment (e.g., P2P)
  - Multiple-to-one streaming
  - Heterogeneous, failure-prone suppliers
  - Dynamic network conditions
- Challenge
  - Achieve and maintain full-quality
- Our Solution
  - CollectCast: based on tomography

2. CollectCast
- Infer approximate network conditions (avail bw, loss, topology)
- Select best peers from a candidate set
- Adaptive assignment of rate and data to suppliers
- Seamless supplier switching to maintain full quality

3. Inference
- Adapt tomography techniques, e.g.,
  - Not interested in “exact” avail bw, rather, can a path support aggregate rate from sullying peers?
  - Probe with real (movie) data!
  - Peers are weak: coordinate probing from multiple peers
- Result
  - Topology annotated with segment-wise loss and avail bw

4. Suppliers Selection
- Find suppliers \( (P_{actv}) \) that:
  - \( \max E [\sum_p G_p R_p] \)
  - Subject to \( a_p R_p \leq \sum_p R_p \leq a_q R_q \)
- \( G_p \): How good peer \( p \) is for this session:
- \( \alpha \): weight based on avail bw and level of sharing

5. Rate/ Data Assignment
- Assign rate/data to suppliers with adaptive FEC
  - Pre-encode segments, \( \text{FEC}(\alpha_u) \)
  - Send at \( \alpha R_0 \) to tolerate current aggregate loss rate
  - Typical: \( 1 \leq \alpha \leq \alpha_u = 1.25 \)
- \( \hat{R}_p \): Assigned Rate
- \( \hat{D}_p \): Assigned Data

6. Adaptation
- Peer failure/degradation \( \Rightarrow \) switch suppliers
  - Update topology, labels
  - Solve the maximization problem
  - Note: keep the good peers that you already have!
- Network fluctuations
  - Adjust \( \alpha \) (loss tolerance level)
    - Reduce redundancy if network is fine
    - Increase, otherwise
  - If new \( \alpha \) is greater than what current peers can support, add/replace peer(s)

7. Overhead
- Communication overhead
  - We use real data for probing \( \Rightarrow \) little overhead!
  - Larger receiver buffer, though (order of Mbytes)
- Processing overhead
  - To run the estimation procedures and construct the topology
  - Not a big concern (order of milliseconds)
- Frequency of update
  - Internet path properties (loss, bw, delay) exhibit a relative constancy, at least in order of minutes \([Zhang et al., IMW'01]\)

8. Evaluation: Sample Results
- Setup
  - Large topology, Markov losses, random avail bw
  - Peers fail
  - Select peers using
    - CollectCast (tomography)
    - E2E (no notion of shared segments)
    - Random
  - Measure aggregate received rate
- How much do we gain?
- How many candidates?
- Candidate set size depends on reliability

9. Application
- PROMISE—P2P Streaming Using CollectCast
  - Integrated Pastry, CollectCast
- More Info at
  - www.cs.purdue.edu/~mhefeeda
- Support
  - NSF grant ANI-0219110