Server-based Inference of Internet Performance

Network Performance Measurement

 Authors

- Venkata N. Padmanabhan
  - Ph.D. in Computer Science from the University of California at Berkeley in September 1998
  - Researcher at Microsoft Research
- Lili Qiu
  - Ph.D. in Computer Science from Cornell
  - Researcher at Microsoft Research
  - Interested in Web performance, Network measurement, Adaptive applications, Packet Classification, Wireless network, TCP, active queue management, and congestion control
- Helen Jiahe Wang
  - Ph.D. in Computer Science from the University of California at Berkeley
  - Researcher at Microsoft Research
  - Interested in networking, protocol architectures, large scale wide-area distributed computing and communication systems, and mobile computing

 Introduction

- Goal:
  - Investigate ways to infer Internet performance by passively monitoring existing network traffic
  - Develop techniques to infer performance of interior links
- Focus
  - Packet loss rate - direct indicator of network congestion
  - How well does loss rate correlate with topological distance between server and client?
  - How stable is the loss rate over time?
  - How strong is the spatial locality in loss rate?

Related Work

- Active studies – inject traffic into the network
  - May alter link characteristics
- Passive studies – analyze existing traffic
  - Existing traffic may not contain enough data to make an inference
- Previous studies mostly focused on throughput
Experimental Setup and Methodology

- microsoft.com
- tcpdump to do packet capture
- Packet sniffer captured only the headers of TCP packets
- traceroute to determine the network path to each of the clients
- Packet loss detected during retransmission
- Loss rate for client = (packets retransmitted)/(total packets sent)

Analysis of End-to-End Loss Rate

- Correlation between Topological Distance and Loss Rate
  - Route hop count - determined by traceroute
  - AS hop count - computed by looking at AS number for each router
  - Address prefix (AP) hop count - determined by looking at BGP prefix for each router

Analysis of End-to-End Loss Rate (Correlation between Topological Distance and Loss Rate)

- Little correlation between loss rate and hop count
  - Hop count - not reliable indicator of packet loss rate
  - Links are not equal
  - Poor end-to-end performance caused by a few lossy links.

Analysis of End-to-End Loss Rate

- Temporal Locality of Loss Rate
  - Loss rates partitioned into categories
    - (0-0.5%, 0.5-2%, 2-5%, 5-10%, 10-20%, 20+%)  
  - How long the loss rate remains in the same category
Analysis of End-to-End Loss Rate

(Temporal Locality of Loss Rate)

- Loss rate is stable on the time scale of several minutes.
- Lossiness of network links is likely to persist for a significant length of time.

Analysis of End-to-End Loss Rate

(Spacial Locality of Loss Rate)

- Some spatial locality especially at subnet level.
- Shared cause for packet loss.
- Most often the cause of packet loss is a non-shared link.
Passive Network Tomography

- Server transmits data to distributed set of clients
- Network path to each client is known (traceroute)
- Assume each link has constant loss rate
- Treat linear sections of network path as “virtual link”

Given $M$ clients and $N$ links, we have $M$ constraints (corresponding to each server-client path) defined over $N$ variables (corresponding to the loss rate of the individual links). For each client $C_i$, there is a constraint of the form $1 - \prod_{j \in C_i} (1 - l_j) = p_i$ where $C_i$ is the set of links on the path from the server to client $C_i$, $l_j$ is the loss rate of link $j$, and $p_i$ is the end-to-end loss rate between the server and client $C_i$. There is not a unique solution to this set of constraints if $M < N$, as is often the case.

Random Sampling

- Algorithm:
  - Assign a loss rate of 0 to each link in the tree
  - Pick the loss rate $l_j$ of the link $j$ to be a random number between 0 and $\max_{i \in C_j}$
  - The link loss rate is bounded by $\min_{i \in C_j}$
  - Residual loss rate $= 1 - \prod_{j \in C_i} (1 - l_j)$
  - Repeat the procedure to compute the residual loss rate for each client
  - Iterate $R$ times to produce $R$ random solutions
  - Order in which the links are picked matters
  - Susceptible to estimation errors

Linear Optimization

- Define the problem as LP
- Minimize $\sum_l l_i + \sum_j S_j$
- Subject to $\sum_{i \in C_j} l_i + S_j = r_j$
- $l_i \geq 0$, $S_j \geq 0$ and $r_j \geq -S_j$

- Depends on client loss rates to be computed
- Large number of packets has to be sent
- Solution would be different if the objective function is modified

Bayesian Inference using Gibbs Sampling

- Bayesian inference determines the posterior distribution of $\theta$, $P(\theta | D)$ based on the observed data $D$
- Inference based on prior distribution $P(\theta)$ and likelihood $P(D | \theta)$
- Use Markov Chain Monte Carlo method and Gibbs sampling
- $D = \bigcup_{i \in U} c_i (t_i)$
- $\theta = l_i \cup \min_{j \in C_i} p_j$

- Algorithm:
  - Initially arbitrary assign link loss rates
  - At each step pick a link and numerically compute Posterior distribution of loss rate:
    $P(l_i | D, c_i (t_i)) = \frac{P(D | l_i, c_i (t_i)) P(l_i)}{\int P(D | l_i, c_i (t_i)) P(l_i) d\theta}$
  - Cycle through all the links and assign each a new loss rate
  - After a burn-in period (a few hundred iterations) obtain samples from the distribution $P(\theta | D)$
  - Only requires number of packets sent
Simulation

- Topology – randomly constructed trees
- Assign link loss rates
  - LM1 – good links 0-1%, bad links 5-10%
  - LM2 – good links 0-1%, bad links 1-100%
- Bernoulli case – each packet is dropped with a fixed probability
- Gilbert case – link fluctuates between good (no packets dropped) and bad state (all packets dropped)
- Probability of staying in bad state – 35%
- Other state transition probability is picked to match the link loss rate
- Experiment is repeated 6 times for each configuration

Simulation Results (LM1 & Bernoulli)

Simulation Results (Gibbs)

Simulation Results (Node degree)
Simulation Results

- **Random Sampling**
  - Has best coverage, identifies 90–95% of lossy links
  - High false positive rate
  - Better performance with high node degree
  - Quickest to compute

- **Linear Optimization**
  - Poor coverage, identifies 30–60% of lossy links
  - Low false positive rate, rarely above 5%
  - Lower weight → better coverage, but higher false positive rates

- **Gibbs**
  - Good coverage, identifies over 80% of lossy links
  - Low false positive rate, under 5%
  - Hard to compute

Real Topology

- **LMS Bernoulli loss model with different settings for f**
  - 123156 clients

Real Topology (random sampling)

Internet Results

- **Validation**
  - Check consistency in the inferences made by the three techniques
    - Overlap is significant, especially in Gibbs and Random sampling
  - Look at characteristics of inferred lossy links
    - Most of the links are non-shared
    - Limited degree of spatial locality
  - Examine if clients downstream of inferred lossy link experience high loss rates
Trace Driven Validation

- Clients partitioned into two groups (tomography set and validation set)
- BGP address prefix clustering
- Apply inference techniques to tomography set to identify lossy links
- For each identified lossy link, check the loss rate experienced by clients in validation set
  - If high loss rate – inference is correct
  - If not – count as false positive
- Can be applied to shared links only

## Conclusion

- End-to-end packet loss
  - Correlates poorly with topological distance
  - Remains stable for several minutes
  - Has a limited degree if spatial locality
- Developed and evaluated three techniques for passive network tomography
- Most of the links identified as lossy are non-shared