



## Server-based Inference of Internet Performance

---

### Network Performance Measurement

1



## 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

2



## Introduction

- Goal:
  - Investigate ways to infer the 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?
  - Passive Network Tomography
    - Identify interior lossy links by passively observing the existing traffic

3



## 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

4

## 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 =  $(\text{packets retransmitted}) / (\text{total packets sent})$

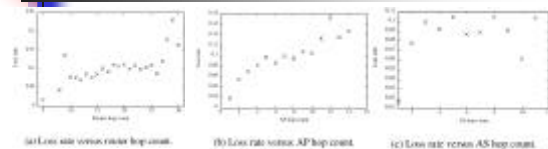
5

## 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

6

## 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.

7

## 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

8

## Analysis of End-to-End Loss Rate (Temporal Locality of Loss Rate)

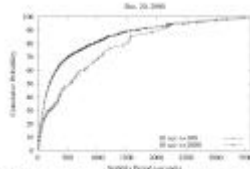


Fig. 2. CDF of the time period in which a host's loss rate remains in the same loss category.

- 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

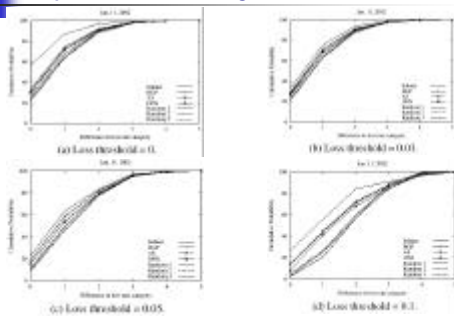
9

## Analysis of End-to-End Loss Rate

- Spatial Locality of Loss Rate
  - Clients clustered (subnet, AP, AS, DNS, randomly)
  - Pick two groups with three clients from each cluster
  - Compare average loss rates
  - Compare number of loss categories the two groups are away from each other
  - Loss threshold

10

## Analysis of End-to-End Loss Rate (Spatial Locality of Loss Rate)



11

## Analysis of End-to-End Loss Rate (Spatial 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

12

## 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  $C_j$ , there is a constraint of the form  $1 - \prod_{i \in T_j} (1 - l_i) = p_j$  where  $T_j$  is the set of links on the path from the server to client  $C_j$ ,  $l_i$  is the loss rate of link  $i$ , and  $p_j$  is the end-to-end loss rate between the server and client  $C_j$ . There is not a unique solution to this set of constraints if  $M < N$ , as is often the case.

13

## Random Sampling

- Algorithm:
  - Assign a loss rate of 0 to each link in the tree
  - Pick the loss rate  $l_i$  of the link  $i$  to be a random number between 0 and  $l_i^{(max)}$ 
    - The link loss rate is bounded by  $l_i^{(max)}$
  - Residual loss rate =  $1 - \prod_{i \in T_j} (1 - l_i)$
  - 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

14

## Linear Optimization

- Define the problem as LP
  - Minimize  $\theta \sum_i l_i + \sum_j S_j^*$
  - Subject to
 
$$\begin{cases} \sum_{i \in T_j} l_i + S_j^* = p_j \\ l_i \geq 0 \\ S_j^* \geq S_j \text{ and } S_j^* \geq -S_j \end{cases}$$
- 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

15

## 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
 
$$D = \bigcup_j (s_j, p_j) \quad \theta = l_i = \bigcup_{i \in E} l_i \quad P(D | \theta) = \prod_{j \in \text{clients}} (1 - p_j)^{n_j} p_j^{c_j}$$
- Algorithm
  - Initially arbitrary assign link loss rates
  - At each step pick a link and numerically compute Posterior distribution of loss rate:  $P(l_i | D, \{l_i\}) = \frac{P(D | \theta_i)}{\sum_{l_i} P(D | \theta_i) W_i}$
  - 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  $P(l_i | D)$
- Only requires number of packets sent

16

## 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

17

## Simulation Results (LM1 & Bernoulli)

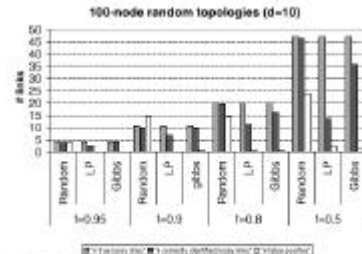


Fig. 6. Varying  $f$ : 100-node random topologies with maximum degree = 10.

18

## Simulation Results (Gibbs)

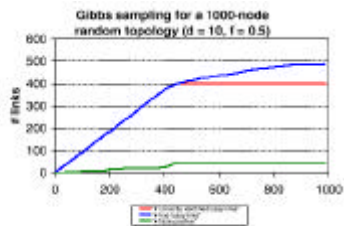


Fig. 10. The performance of Gibbs sampling when the inferences are rank ordered based on a confidence estimate. (1000-node random topology, maximum degree = 10, and  $f = 0.5$ )

19

## Simulation Results (Node degree)

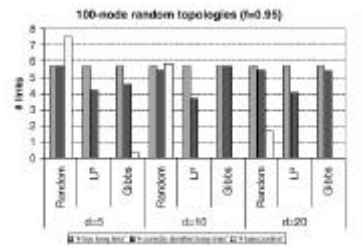


Fig. 11. Varying degree: 100-node random topologies with  $f = 0.95$ .

20

## Simulation Results

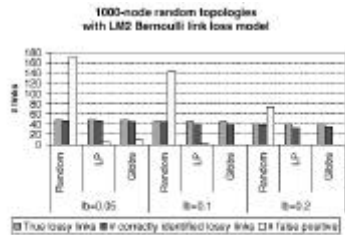


Fig. 12. A LM2 Bernoulli loss model for 1000-node random topologies with maximum degree = 10 and  $f = 0.95$ . We vary the loss threshold  $l$ , and only the links with loss rate higher than  $l$  are considered lossy.

21

## 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  $\rightarrow$  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

22

## Real Topology

- LM1 Bernoulli loss model with different settings for  $f$ , 123166 clients

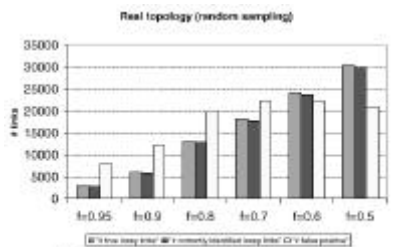


Fig. 17. Real topology from the Dec 2000 tracroute.

23

## 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

24

## 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

25

$L_i$	Method	$t$	$N_i$	$N_c$
4%	Rand	1000	5	5
	Rand	500	5	4
	LP	1000	8	5
2%	LP	500	11	6
	Rand	1000	11	10
	Rand	500	14	13
1%	LP	1000	22	14
	LP	500	24	20
	Rand	1000	24	17
	Rand	500	23	19
1%	LP	1000	46	28
	LP	500	106	77

TABLE II  
TRACE-DRIVEN VALIDATION FOR RANDOM SAMPLING AND LINEAR OPTIMIZATION.

26

## Conclusion

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

27