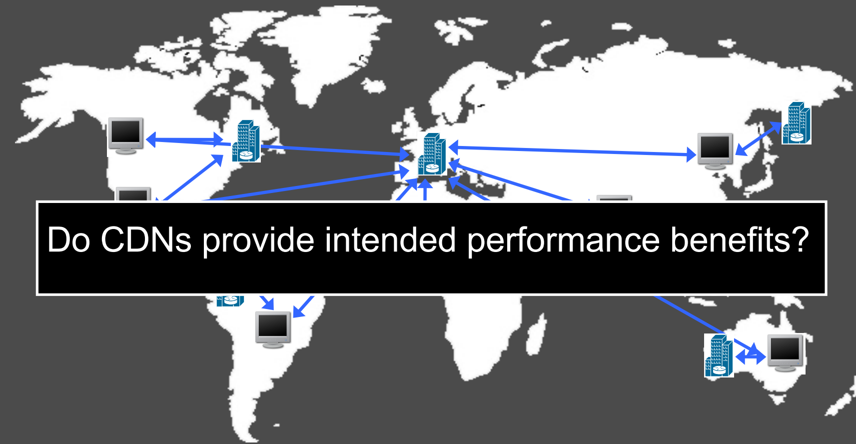
Moving Beyond End-to-End Path Information to Optimize CDN Performance

Rupa Krishnan, Harsha V. Madhyastha, Sushant Jain, Sridhar Srinivasan, Arvind Krishnamurthy, Thomas Anderson, and Jie Gao

Google, University of California San Diego, University of Washington, and Stony Brook University

Motivation for CDNs

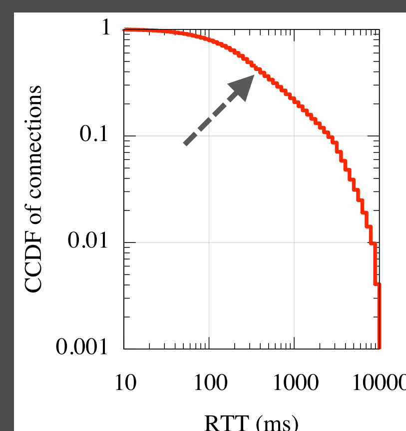


- Problem: Distant clients get poor performance
- Solution: Setup new data centers near clients

Measuring Google CDN

- Tens of CDN nodes serve millions of clients
- Well-engineered and geographically diverse
  - Client redirected to node with least latency
  - 75% of clients have a node within 1000 miles (expected RTT ≈ 20ms)
- Measured RTTs on all connections for a day
  - Passively tracked RTT at TCP connection setup

Is the CDN Effective?



- >400ms RTT on 40% of connections
  - 400ms is more than RTT all the way around the globe

How to Improve Performance?

- Common approach: Add data centers (DCs)
  - Example: Added new DC in Japan to improve performance for clients in Japan
    - Decreased min/avg RTT but not max RTT
- Our approach: Get more out of existing DCs
  - Key question: Why aren't existing DCs providing intended client performance?

Overview

- Identify causes for poor RTTs with CDN
- Develop WhyHigh to aid network admins in troubleshooting inflated RTTs
  - Identify, diagnose, and prioritize problems
- Use WhyHigh to improve Google CDN

Outline

- Motivation
- Problem Statement
- Diagnosing Inflated RTTs
- Developing and Using WhyHigh
- Conclusions

Potential Causes of RTT Inflation

- Three components of end-to-end RTT
  - Transmission delay
  - Propagation delay
  - Queuing delays

Potential Causes of RTT Inflation

- Three components of end-to-end RTT
  - ✗ Transmission delay
  - Propagation delay
  - Queuing delays
- Transmission delay negligible
  - Measurements from small TCP control packets

Potential Causes of RTT Inflation

- Three components of end-to-end RTT
  - ✗ Transmission delay
  - Propagation delay
    - Large distance to CDN node
    - Circuitous route to CDN node
  - Queuing delays

Analysis of RTT Dataset

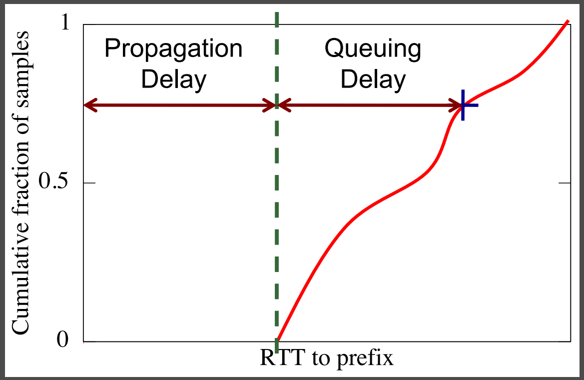
- Annotated every client with
  - Prefix: Granularity of CDN redirection
  - Geographic region (approx. state/province)

Result 1: CDN redirects almost all prefixes to geographically closest node

Potential Causes of RTT Inflation

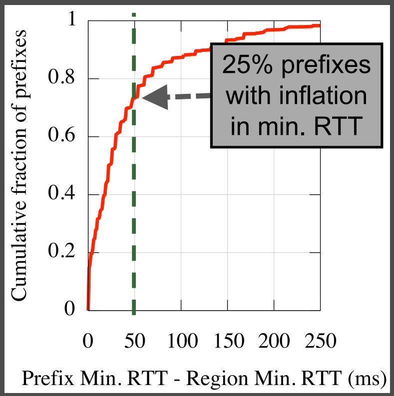
- Three components of end-to-end RTT
  - ✗ Transmission delay
  - Propagation delay
    - ✗ Large distance to CDN node
    - Circuitous route to CDN node
  - Queuing delays

Investigating Propagation Delays



- Minimum RTT to prefix unlikely to have queuing component

Investigating Propagation Delays



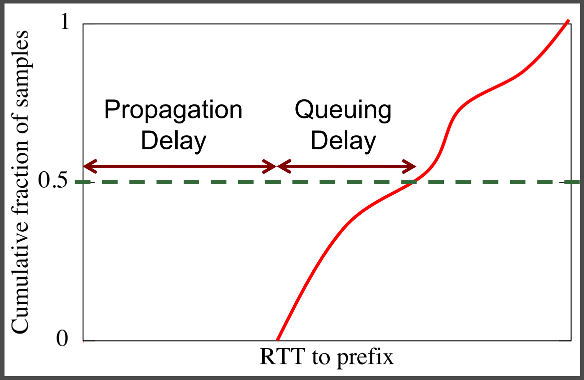
- Evaluate inflation in propagation delay
- Minimum RTT to region considered baseline

Result 2:
Huge disparity in propagation delays to nearby clients

Potential Causes of RTT Inflation

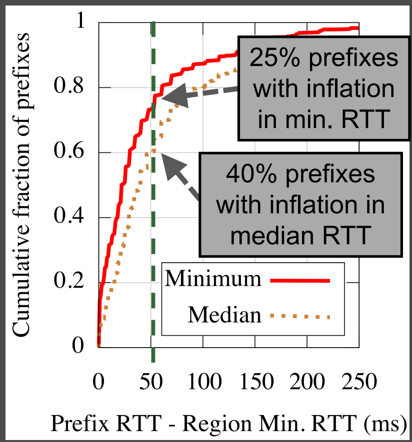
- Three components of end-to-end RTT
  ✗ Transmission delay
  - Propagation delay
    ✗ Large distance to CDN node
    ✓ Circuitous route to CDN node
  - Queuing delays

Investigating Queuing Delays



- Use median RTT to capture queuing delay

Investigating Queuing Delays



- Use median RTT to capture overhead over propagation

Result 3: Large variance in RTTs to a prefix

- What else could explain variance?

Potential Causes of RTT Inflation

- Three components of end-to-end RTT
  - ✗ Transmission delay
  - Propagation delay
    - ✗ Large distance to CDN node
    - ✓ Circuitous route to CDN node
  - ✓ Queuing delays
- How to troubleshoot these problems?
  - We focus on inflation in propagation delay

Outline

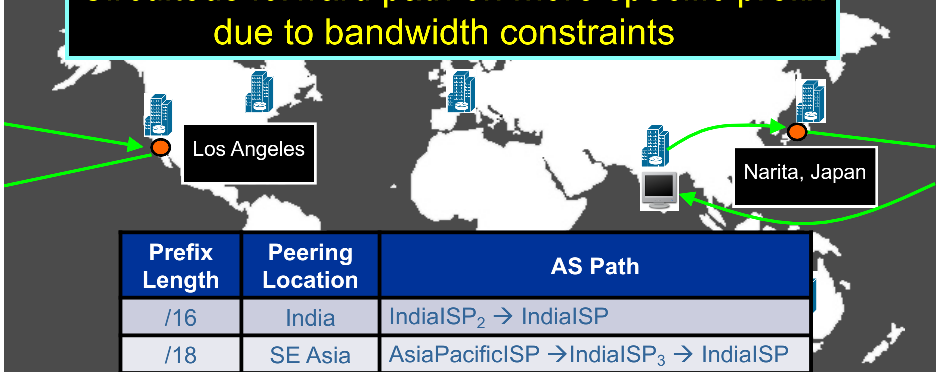
- Motivation
- Problem Statement
- Diagnosing Inflated RTTs
- Developing and Using WhyHigh
- Conclusions

Troubleshooting RTT Inflation

- Main challenge: Too many inflated prefixes
  - Tens of thousands of prefixes to troubleshoot
- Developed WhyHigh to aid network admins
  - Identify
  - Diagnose
  - Prioritize

Use of WhyHigh: Example 1

Circuitous forward path on more specific prefix due to bandwidth constraints



Prefix Length	Peering Location	AS Path
/16	India	IndiaISP$_2$ → IndiaISP
/18	SE Asia	AsiaPacificISP → IndiaISP$_3$ → IndiaISP
/18	Japan	AsiaPacificISP → IndiaISP$_3$ → IndiaISP

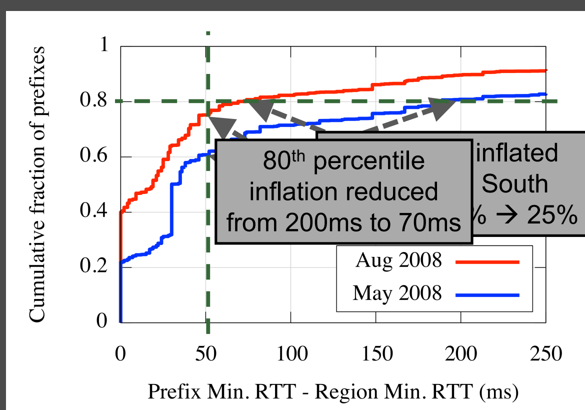
- 4
  - CDN node in India, no Google <-> IndiaISP peering

Use of WhyHigh: Example 2

Circuitous reverse path due to misconfiguration

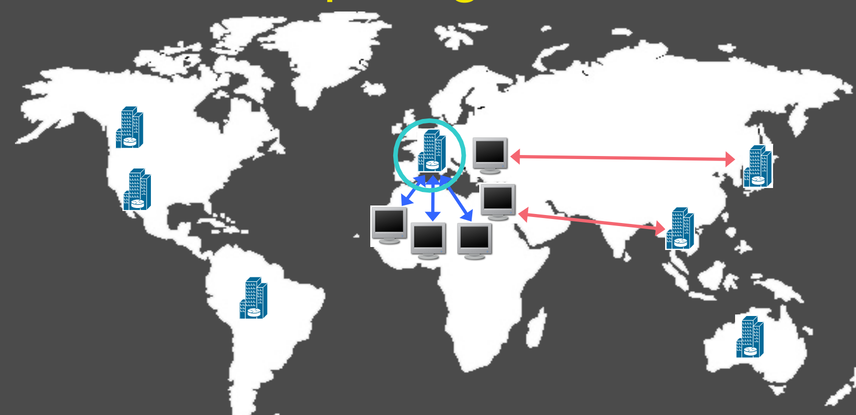
- 100ms RTTs to clients in JapanISP
  - CDN node in Japan, JapanISP peers with Google

CDN Improvement with WhyHigh



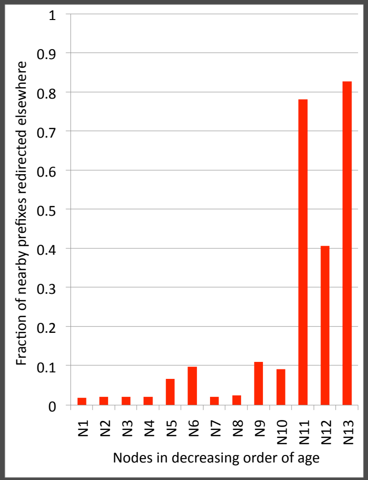
- Significantly improved client RTTs offered by a CDN node in South America

Comparing Nodes



- Metrics for identifying poor node perf.
  - Example: Fraction of nearby prefixes redirected to some other node

Comparing Nodes



- Metrics for node performance
  - Example: Fraction of nearby prefixes redirected to some other node
- Help monitor improvement in new CDN nodes

Conclusions

- Improving CDN performance does not always require adding new nodes
  - Equally important to effectively use and configure existing nodes
- State-of-the-art CDN affected by
  - Circuitous routes to nearby CDN node
  - Queuing of packets
- Developed WhyHigh
  - Used at Google
  - Improved performance of Google CDN

Follow-on Research

- Led to reverse traceroute
  - NSDI 2010 Best Paper
- Increasing use of anycast
  - Example: Bing
- Interest in dynamic changes to redirection
  - React to congestion