Ukairo: Scalable Detour Routing for the Masses

Thom Haddow[*,1] Sing Wang Ho[+] Jonathan Ledlie[†]
Cristian Lumezanu[‡] Moez Draief[+] Peter Pietzuch[*]

[*] Imperial College London, Dept. of Computing
[+] Imperial College London, Dept. of Electrical Engineering
[†] Nokia Research
[‡] Georgia Tech

July 8, 2010

[1] thaddow@doc.ic.ac.uk

Introduction

- The Internet provides a *best-effort* service
  - Does not generally optimise for any particular user-centric metric
  - Measurement shows end-to-end paths generally suboptimal
- Overlay networks are widely deployed to improve Internet properties
  - Adding functionality, improving performance...
  - Feasible to deploy
- Detour routing is a generic form of overlay network
  - Redirecting arbitrary end-to-end traffic via tertiary nodes to change path properties
  - Challenge lies in discovering appropriate *detour* nodes

Motivation

- Detour routing already shown to be effective for latency & availability
  - Easy and cheap metrics to measure
  - Good for real-time applications (VoIP, gaming . . .)
  - Already used commercially (Akamai)
- Bandwidth is potentially much more useful metric to optimise
  - Can improve performance of any bulk-transfer application
  - File transfers, HD media streaming. . .
- However, bandwidth is both difficult and expensive to measure
  - Most work on bandwidth detouring either small-scale or analytical

Overview

- Bandwidth as a metric
- Scope for detouring on the Internet today
- Exploiting detours

Measuring Bandwidth

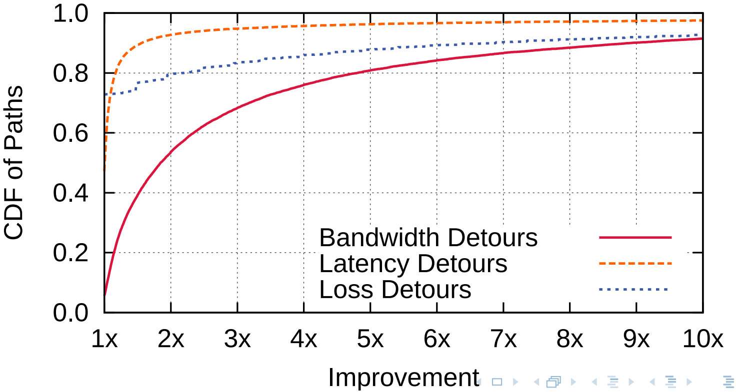
- Dataset: All-pairs Iperf bandwidth (BTC) measurements. 250 nodes, PlanetLab
- Iperf: Measures standard TCP connection
  - **Representative of application performance!**
  - Wastes a lot of bandwidth (∼10MB/path)
- Compared various "lightweight" analytical bandwidth estimation tools
  - Measuring via precisely timed packet bursts
  - Generally very slow. Give little insight into achievable throughput
- Developed our own estimation tool which wastes less bandwidth by detecting when TCP throughput has reached steady-state

Bandwidth Observations

- Bandwidth is elastic
  - Introducing new traffic causes competing flows to back-off
  - Measurements extremely noisy
- Measurements consistent over moderate time intervals
  - Distribution of bandwidth variation in a 90 minute window is the same as over 30 seconds
  - Thus bandwidth detours can remain consistent over time
- **96.6% of paths can potentially benefit from bandwidth detouring!**
  - ie, for the vast majority of node-pairs, there exists another node to which they both have more bandwidth to than each other.

Detouring for Bandwidth

- **Potential for improvement in bandwidth is much greater than for other metrics**
  - Little improvement available from multi-hop detours



Exploiting Detours: IP Detouring

- Previous results based on analysis of measurements. We wish to evaluate real-world large-scale performance.
- Natural approach to building such a system would be to simply redirect IP packet flows via the detour node — **IP detouring**
  - Easily implemented with IP-in-IP tunneling and NAT.
  - Transparent to end nodes. Can be done anywhere en route.

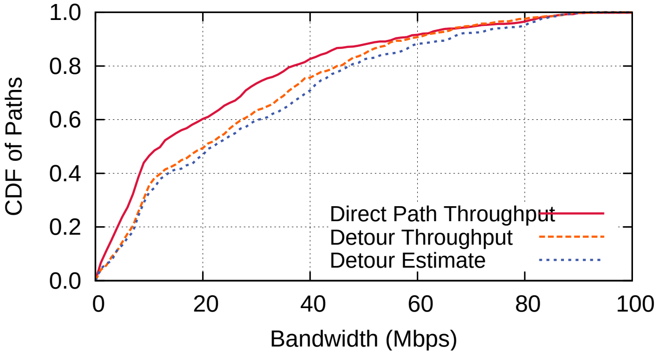
IP Detouring Performance

- End-to-end detour path properties composed from two constituent end-to-end IP paths.
  - $BW_{ABC} = min(BW_{AB}, BW_{BC})$?
- IP detour path may have higher or lower latency than the existing “direct” path
  - Two complete end-to-end paths $\rightarrow$ more hops
  - Changes the loss characteristics, which **severely** affects TCP throughput.
- We have an equation which predicts the throughput of the IP detour path, defined terms of the throughputs and latencies of the constituent paths.

TCP Detouring

- TCP throughput terrible on IP detour paths
- Sidestep by using a separate TCP connection over each of the constituent paths
  - Essentially split-TCP, except split outside rather than inside the network
- Implemented with customised SOCKS proxy and sockisfying library
  - Possible to do transparently in network, but complex
  - Still remains transparent to applications
- Would expect performance of detour paths to be roughly equivalent to the minimum of the two constituent paths...

TCP Detouring Performance



- **Median BW: 12 $\rightarrow$ 21Mbps**
- Bottleneck at around 50Mbps (PlanetLab?)
- 36.7% of paths double in BW **and** improve by $>$ 1Mbps

Detour Discovery

- Detour routing relies upon discovering a good detour node for any given end-to-end path
  - Not scalable to have a complete picture of the network
- There are many effective proposals for discovering detours for latency
  - Previously we have proposed identifying detours for arbitrary paths using AS-path similarity-clustering based on existing known-good detourable paths.
  - This was effective and scalable for latency, but less so for bandwidth.
- **Using latency detours to identify bandwidth detours is no better than random selection.**
  - Need bandwidth specific approaches

Ukairo: Generic Detour Routing Service

- Currently deploying large-scale dynamic platform on PlanetLab.
  - Measuring performance improvement for both intra-overlay applications (eg BT) and the public Internet (eg improving performance to third-party websites)
- Quantifying the extent to which Split-TCP improves paths, as compared to traversing alternative routes.
- Evaluating new detour discovery approaches
  - Latency-detouring has focused on the structural inefficiencies inherent in Internet routing. Approaches have been based on finding and exploiting "better" alternative routes.
  - For bandwidth, we must equally focus on finding an appropriate node for splitting a TCP connection, thus optimising the transport protocol itself.

Conclusions

- Bandwidth is extremely expensive to usefully measure
- Massive potential for bandwidth improvement can be seen through measurement
- Deployed detour platform can achieve a significant proportion of this, employing only PlanetLab hosts.
- Cannot achieve good performance by simply redirecting IP traffic.
- Approaches for latency detour discovery are not effective for finding bandwidth detours

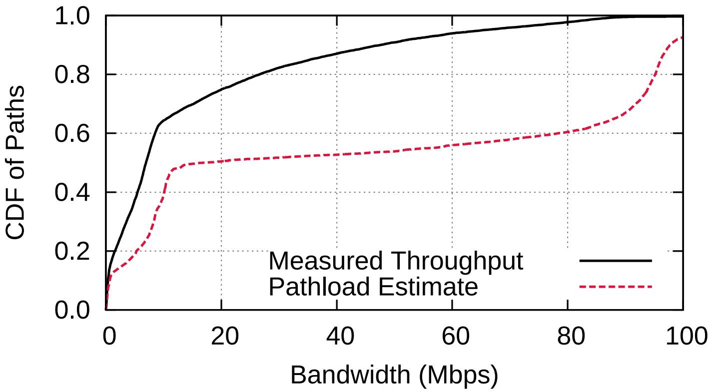
Conclusions

- Bandwidth is extremely expensive to usefully measure
- Massive potential for bandwidth improvement can be seen through measurement
- Deployed detour platform can achieve a significant proportion of this, employing only PlanetLab hosts.
- Cannot achieve good performance by simply redirecting IP traffic.
- Approaches for latency detour discovery are not effective for finding bandwidth detours

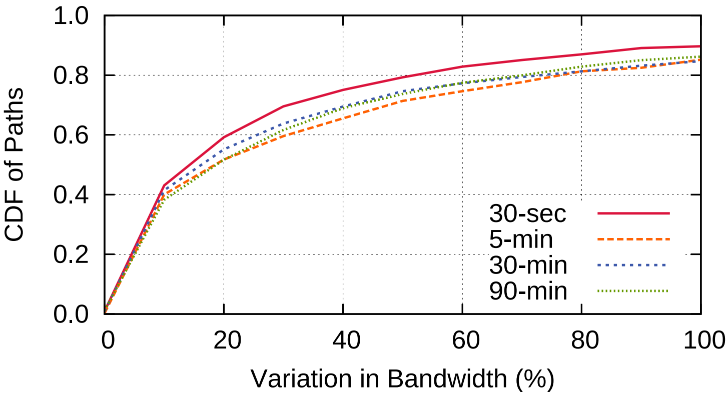
Thanks. Questions?

thaddow@doc.ic.ac.uk

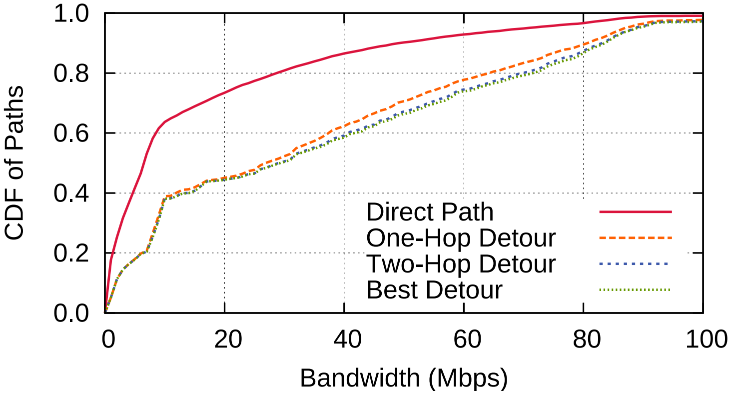
Pathload vs Iperf



Bandwidth Variation



Multi-hop Detours



Throughput Prediction: Equation

$$\mathrm{BTC}_{\mathrm{IP}} \approx \frac{\mathrm{RTT}_1\mathrm{RTT}_2}{\mathrm{RTT}_1 + \mathrm{RTT}_2} \frac{\mathrm{BTC}_1\mathrm{BTC}_2}{\sqrt{(\mathrm{RTT}_1\mathrm{BTC}_1)^2 + (\mathrm{RTT}_2\mathrm{BTC}_2)^2}} \tag{1}$$

Throughput Prediction: Performance