On low-latency-capable topologies, and their impact on the design of intra-domain routing

Nikola Gvozdiev, Stefano Vissicchio, Brad Karp, Mark Handley
University College London (UCL)

Abstract

An ISP's customers increasingly demand delivery of their traffic without congestion and with low latency. The ISP's topology, routing, and traffic engineering, often over multiple paths, together determine congestion and latency within its backbone. We first consider **how to measure a topology's capacity** to route traffic without congestion and with low latency. We **introduce low-latency path diversity** (LLPD), a metric that captures a topology's flexibility to accommodate traffic on alternative low-latency paths. We explore to what extent 116 real backbone topologies can, regardless of routing system, keep latency low when demand exceeds the shortest path's capacity. [...]

Abstract

[...] We find, perhaps surprisingly, that **topologies with good LLPD are precisely those where routing schemes struggle to achieve low latency without congestion**. We examine why these schemes perform poorly, and **offer an existence proof that a practical routing scheme can achieve a topology's potential** for congestion-free, low-delay routing. Finally we **examine implications for the design of backbone topologies** amenable to achieving high capacity and low delay.

Topologies

Internet Topology Zoo (116 topologies)

Finding: ⬆ # of low latency paths ➡ ⬇ low-latency delivery

Comparing: B4, **LDR**, MinMaxK10, MinMax

Not an ideal way to achieve it: Overload a link ➡ change to another link

Unlocking the potential

how to measure a topology’s capacity

A dedicated 1gbps link can be more interesting than an overloaded 100gbps link

d_a/d_s ➜ path stretch

  d_a ➜ delay on alternative path

  d_s ➜ delay on shortest path

Suggested: 40% (1.4)

Alternative Path Availability

APA = paths.map { a -> delay(a)/delay(s) }.filter { it <= 1.4 }.size / paths.size

  range: [0, 1]

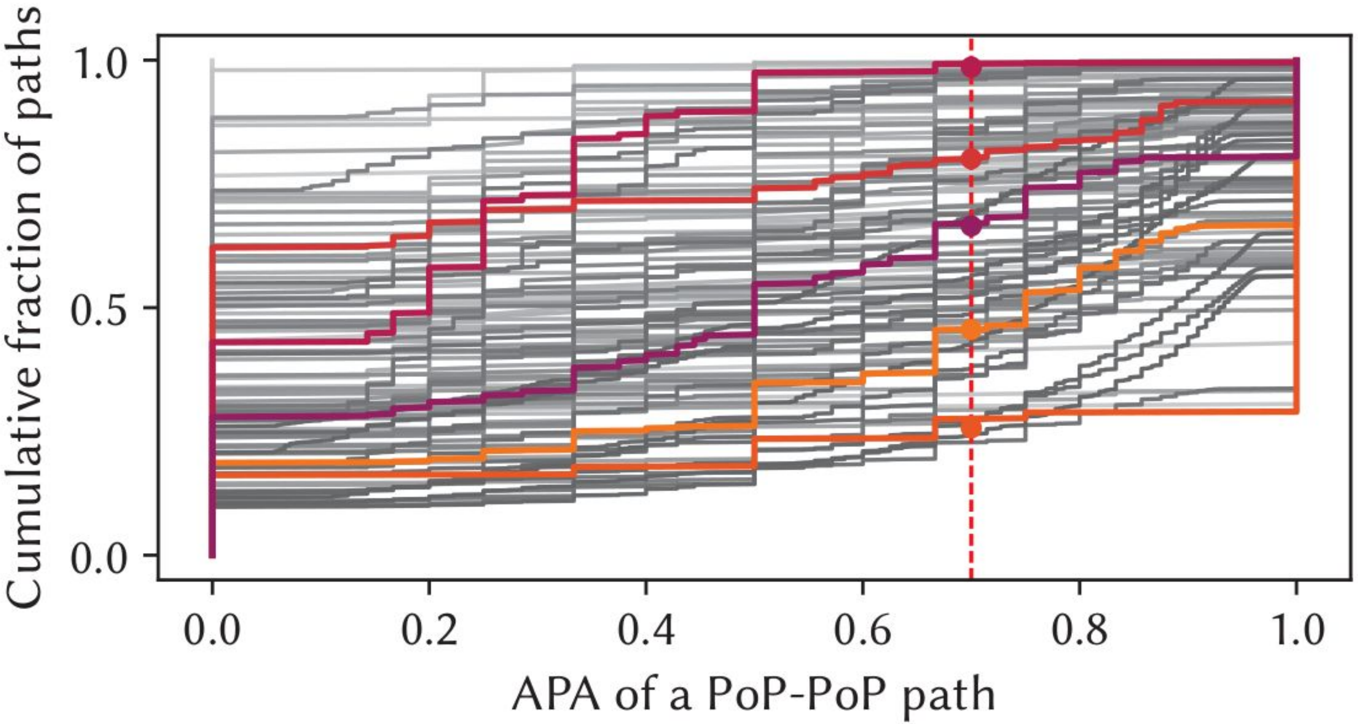


Figure 1: CDF curves of APA for all networks, given path stretch limit of 40%. Five random curves are highlighted.

PoP = points of presence

Low-Latency Path Diversity

All those curves seems like a trend...

$$LLPD = \frac{\text{number of PoP pairs with APA} \geq 0.7}{\text{total number of PoP pairs}}$$

...and indicates how much room for routing change exists on the network.

LLPD close to 1 ➔ can be routed around without excessive delay
LLPD close to 0 ➔ tree-like topology

Right. Path diversity exists, but **how do we use them?**

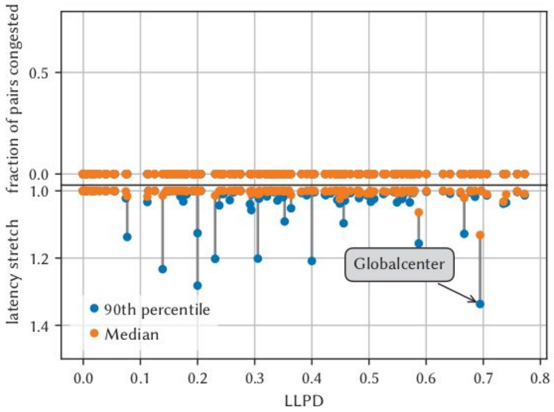
Path diversity is hard to use

Shortest path (link costs = delay) ➜ cannot fully make use of high LLPD networks

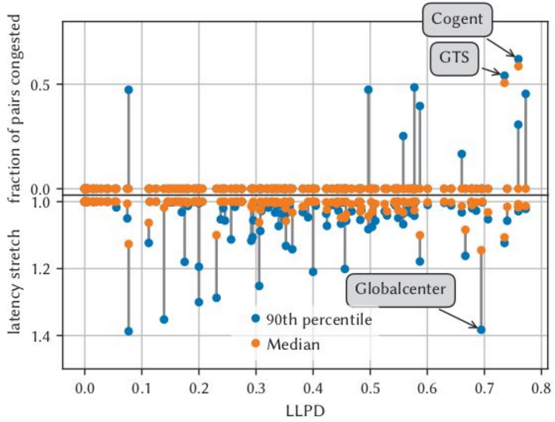
Latency optimality ➜ able to route all traffic without excessive delay stretch

Greedy low latency routing (B4) ➜ doesn’t meet optimal latency on all cases

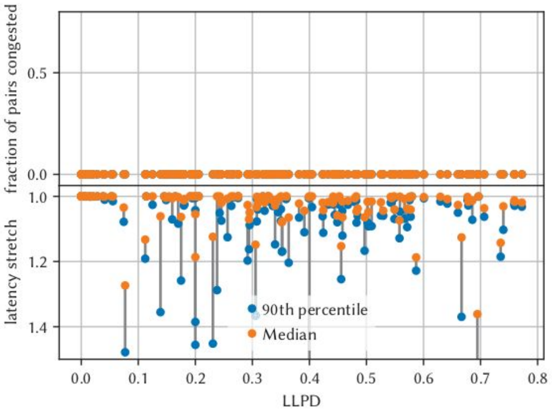
MinMax based routing ➜ Insufficient: needs a larger stretch than others and, when optimization was attempted, congestion was found



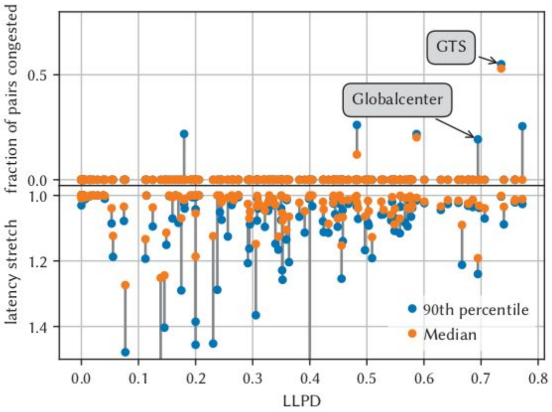
(a) Optimal latency



(b) B4



(c) MinMax



(d) MinMax K=10

Figure 4: Effects of active routing on congestion and delay.

Using path diversity

Consider that your path usage can grow.

Headroom vs latency: enemies

for Google's WAN, 10% is enough

Algorithm 1: Predicting next minute's mean level.

$prev\_value$ // Value measured last minute
$prev\_prediction$ // Value predicted last minute
$decay\_multiplier \leftarrow 0.98$ // 2% decay when level drops
$fixed\_hedge \leftarrow 1.1$ // 10% hedge against growth
$scaled\_est \leftarrow prev\_value * fixed\_hedge$;
if $scaled\_est > prev\_prediction$ **then**
    $next\_prediction \leftarrow scaled\_est$;
else
    $decay\_prediction \leftarrow prev\_prediction * decay\_multiplier$;
    $next\_prediction \leftarrow max(decay\_prediction, scaled\_est)$;

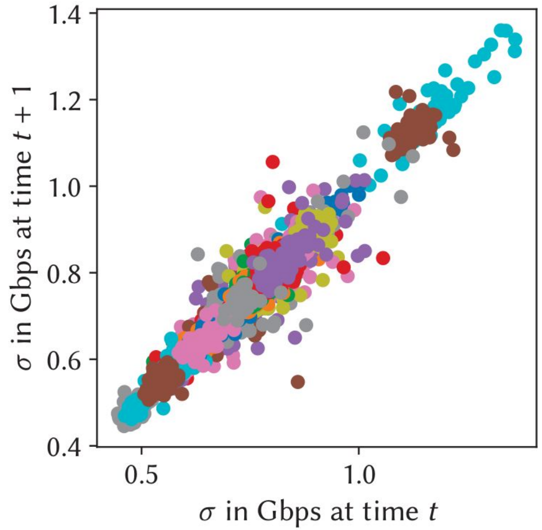


Figure 10: Minute to minute change of the standard deviation of traffic rate.

secondary goal: minimize latency

tie-breaker: minimize delay stretch

primary goal: avoid congestion

if congestion is unavoidable: minimize total overload

$$\mathbf{min} \sum_{a} n_a \sum_{p \in P_a} x_{ap}\left(d_p + \frac{d_p M_1}{S_a}\right) + M_2 O_{max} + \sum_{l} O_l$$

$$\sum_{a} \sum_{p \in P_a} x_{ap} B_a < C_l O_l \quad \forall l \in L$$

per-link overload definition

$$1 \leq O_l < O_{max} \quad \forall l \in L$$

maximum overload definition

$$\sum_{p \in P_a} x_{ap} = 1 \quad \forall a \in A$$

all the aggregates must be routed

$a \rightarrow$ aggregate $B_a \rightarrow$ bandwidth of aggregate $n_a \rightarrow$ flows in aggregate

$p \rightarrow$ path $d_p \rightarrow$ path delay $S_a \rightarrow$ shortest path delay $l \rightarrow$ link $C_l \rightarrow$ link capacity

$P_a \rightarrow$ possible paths for aggregate $x_{ap} \rightarrow$ fraction of a on path $O_l \rightarrow$ overload of link

Figure 12: Linear Program for latency optimization.

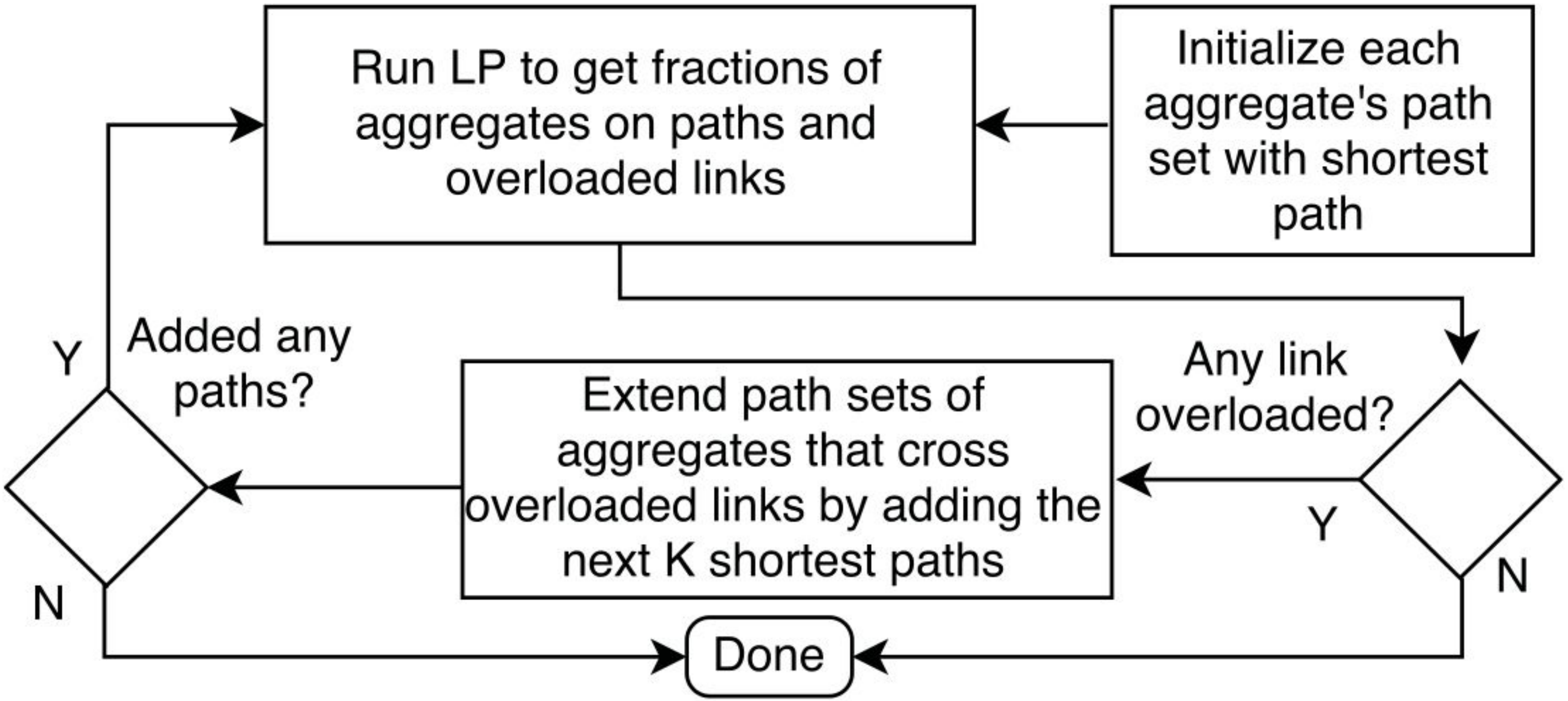


Figure 13: Obtaining paths and per-path aggregate fractions, assuming each aggregate's demand is known.

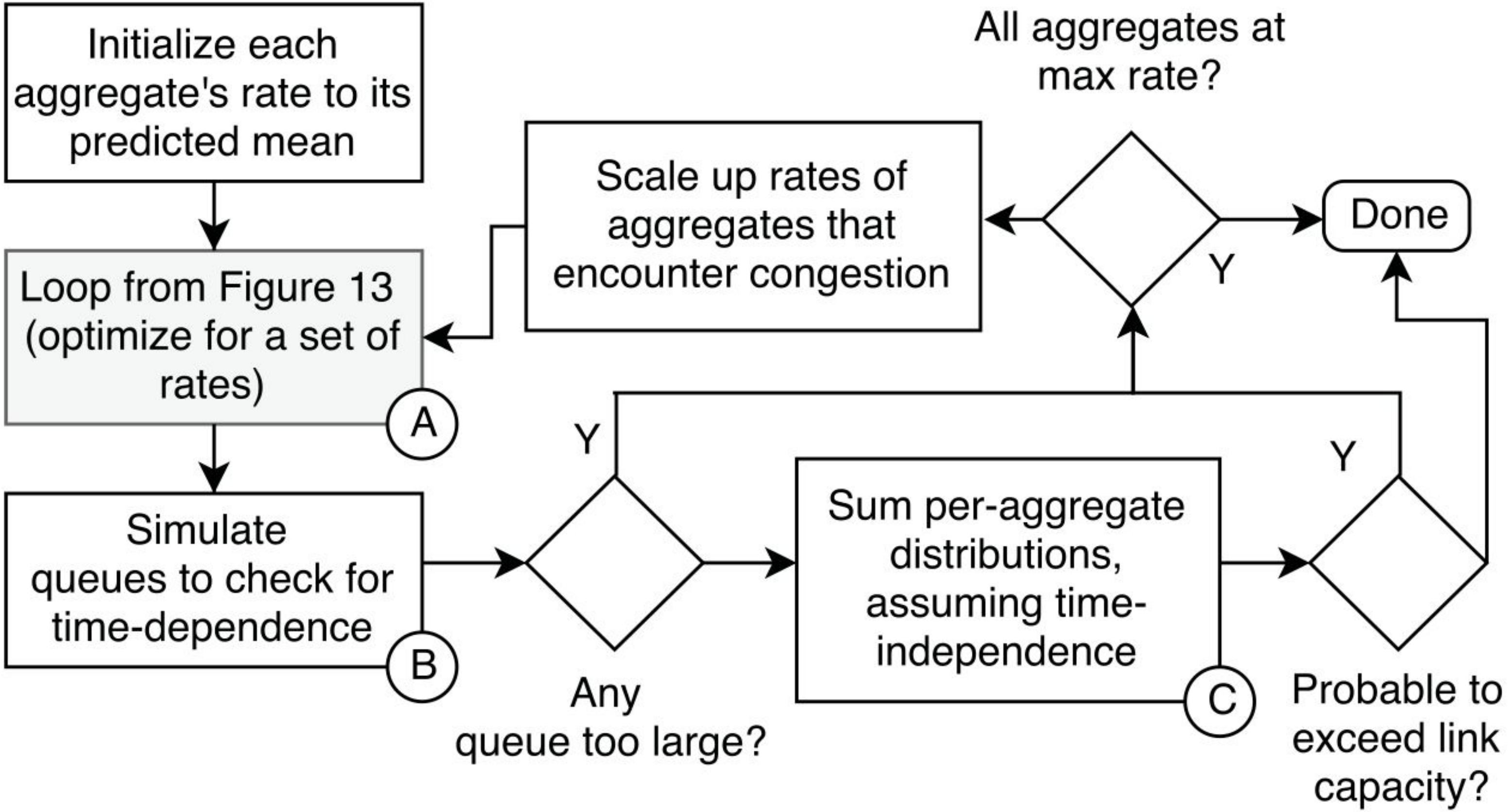
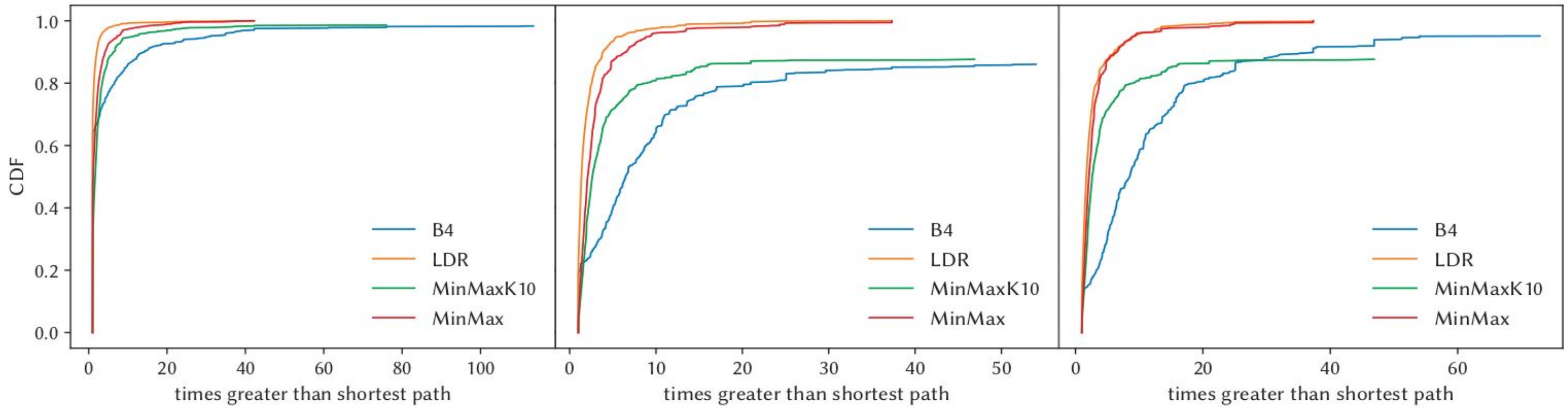


Figure 14: Iteration to assess statistical multiplexing.

As result...



(a) LLPD < 0.5, no headroom

(b) LLPD > 0.5, no headroom

(c) LLPD > 0.5, 10% headroom

Figure 16: Maximum path stretch

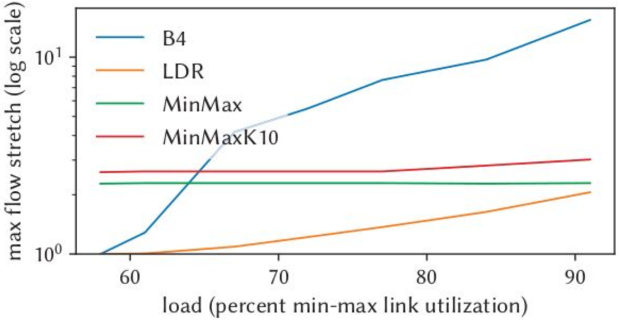


Figure 17: Effect of load on median latency stretch

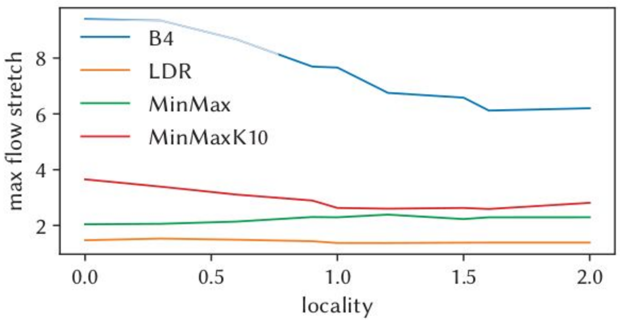


Figure 18: Effect of locality on median latency stretch

Discussion

The Topology Zoo ➜ Not the most modern topologies. Also tested on Google's topology and didn't perform better than their internal B4 implementation.

Does routing influence topology? ➜ Cannot affirm, but 2 ISP topologies tried optimizing their network for the algorithm they would use after the network is ready.

LLPD applicability ➜ *retrospectively* assess path diversity; not *prospectively*.

Traffic classes ➜ not all flows are equal; but they were deemed so. It's not hard to change.