Fibbing in action: On-demand load-balancing for better video delivery

Olivier Tilmans* , Stefano Vissicchio* , Laurent Vanbever† , Jennifer Rexford‡

* Université catholique de Louvain, † ETH Zürich, ‡ Princeton University

www.fibbing.net

www.github.com/Fibbing

Flash crowds cause service disruption

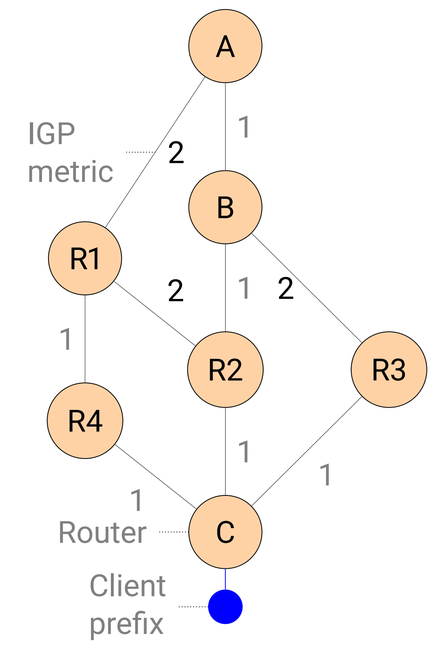
- Video delivery services require good network performance and suffer from flash crowds [1]: transient, localized, surges of traffic.
- Protecting the services at the network level against these surges is hard due to their short-lived nature:
  1. Traditional traffic engineering techniques [2] perform poorly;
  2. Over-provisioning is expensive.

Fibbing lets networks handle flash crowds

- Fibbing [3] can change the behavior of networks within a single IGP convergence.
- Fibbing provides the two required primitives to implement the optimal solution to the min-max link utilization problem [4]:
  1. Programming multiple paths on a per-destination basis;
  2. Enforcing uneven load-balancing [5] among these paths.

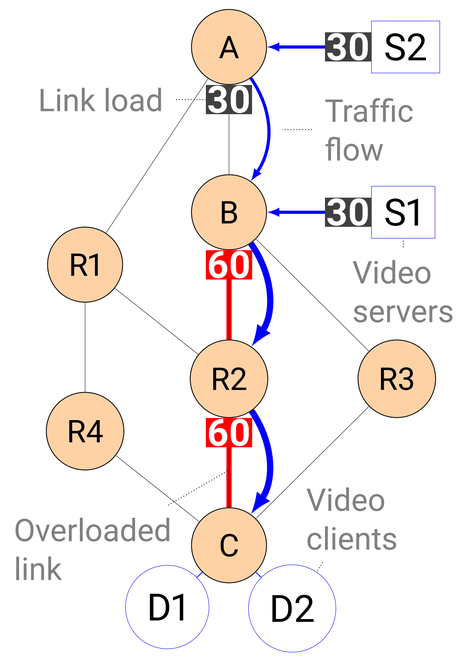
Programming ECMP to decrease the maximal link load using Fibbing

Control-plane



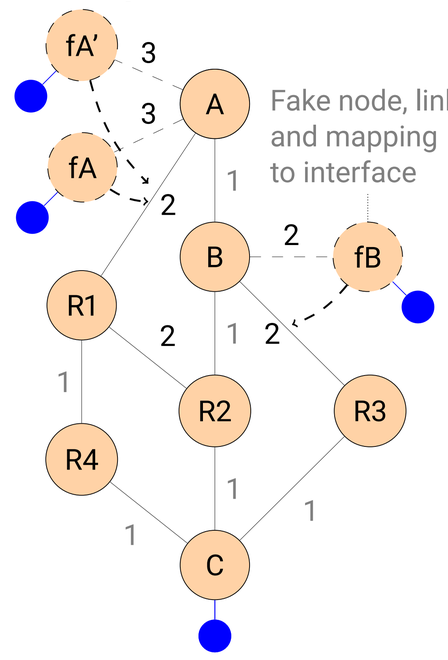
The chosen IGP metrics cause the shortest paths starting at router A and B to overlap along B–R2–C.

Data-plane



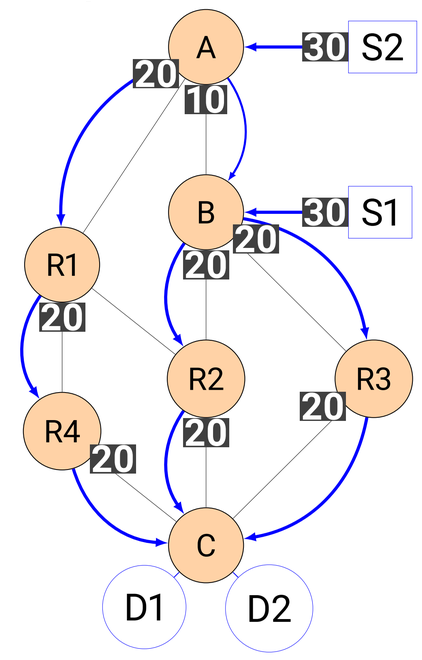
For a high enough demand, the data-plane traffic from the servers to the clients could overload these links.

Control-plane



By augmenting the topology with fake elements, Fibbing creates additional equal-cost paths towards the clients.

Data-plane



The traffic uses all paths, causing uneven load-balancing at router A and decreasing the maximal link load.

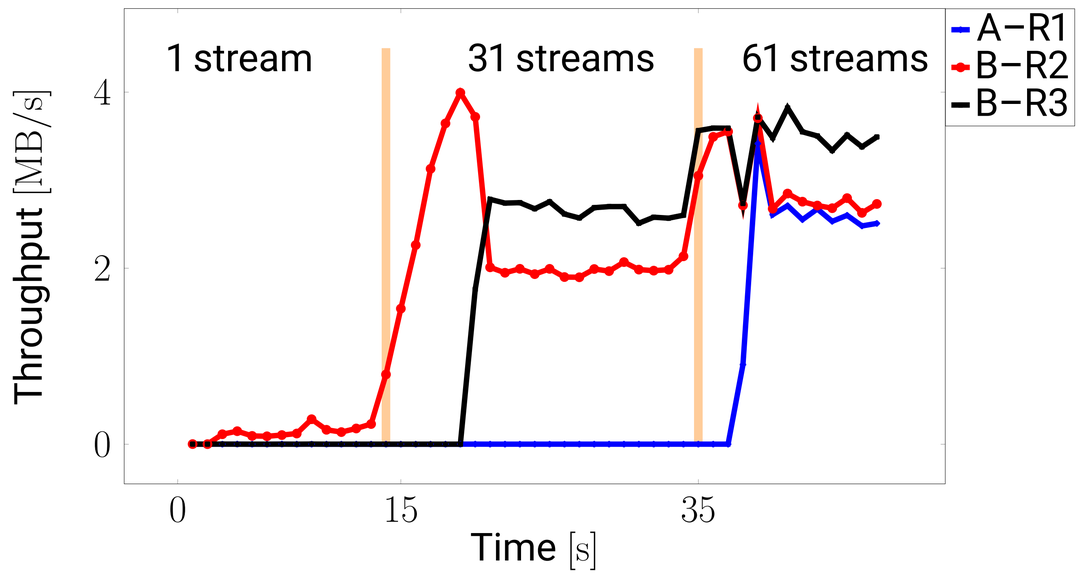
Experiment setup

- We initially have 1 video stream from S1 to D1.
  - At time $t = 15s$, we start 30 new streams from S1 to D1.
  - At time $t = 35s$, we start 30 streams from S2 to D2.
- Servers notify the controller when they start/stop serving a new stream to a client.

Controller behavior

- The controller detects flash crowds using SNMP queries.
- The controller adds a new equal-cost path towards each client prefix such that:
  1. The new path is the shortest link-disjoint path;
  2. The splitting ratio is set to decrease the maximal link load.

Experiment Results



- Additional links are used in response to the traffic increase.
- Introducing uneven load-balancing at router A causes all links to have a similar load.
- The video playbacks on the clients are smooth.

[1] I. Ari, B. Hong, E. L. Miller, S. A. Brandt, and D. D. E. Long, "Managing flash crowds on the internet," in *11th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer Telecommunications Systems*, 2003.
[2] N. Wang, K. H. Ho, G. Pavlou, and M. Howarth, "An overview of routing optimization for internet traffic engineering," *IEEE Communications Surveys & Tutorials*, 2008.
[3] S. Vissicchio, O. Tilmans, L. Vanbever, and J. Rexford, "Central control over distributed routing," in *Proceedings of the 2015 ACM Conference on SIGCOMM*, 2015.
[4] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network flows: Theory, algorithms, and applications*. Prentice Hall, Inc., 1993.
[5] M. Chiesa, G. Kindler, and M. Schapira, "Traffic engineering with equal-cost-multipath: An algorithmic perspective," in *INFOCOM - IEEE Conference on Computer Communications*, 2014.