A System for Coordinated Network-wide Redundancy Elimination

Ashok Anand†, Vyas Sekar‡, Aditya Akella†
†University of Wisconsin-Madison, ‡Carnegie Mellon University

Motivation: Today, an increasing number of end-networks are deploying redundancy elimination (RE) solutions to improve their WAN performance. The success of such deployments has motivated researchers, equipment vendors (e.g.,Cisco, Riverbed) and ISPs to explore the potential of *network-wide* RE. Recent work [1] has shown the benefits of supporting RE as a *primitive IP-layer service* on network routers. In similar vein, network equipment vendors have highlighted the network-wide support for content caching as a key focus area. Supporting RE at the network-level socializes the benefits of RE to all end-to-end flows and also allows ISPs to better handle bandwidth-intensive workloads.

While the problem of RE has been well studied in the context of point deployments (e.g. over a single WAN link of an enterprise network), there has been relatively little work on how to best design network-wide RE deployments. Extending such single vantage point solutions to the network-wide case, as proposed by Anand et. al [1], does not take into account the resource constraints on RE devices. This severely constrains the benefits that end applications and ISPs can derive from network-wide RE.

In this work, we explore how to build effective and practical network-wide RE systems. We present Decor, a coordinated system-wide architecture for in-network RE. Decor allows packets to be decoded at routers multiple hops away from the router where packet was encoded. This allows us to utilize resources from different routers for decoding. Decor takes into account the ISP's objectives (e.g., network-wide footprint reduction) and the resource constraints of different RE devices, to optimally allocate caching and decoding responsiblities. While our current focus has primarily been on RE in ISPs, our design can be more broadly applied to data-center and multi-hop wireless networks.

Design and Implementation: We focus our design on three key elements: ingress, interior nodes and a central configuration model (Figure 1). Ingress nodes encode packets with respect to earlier seen packets in the cache. Interior routers lookup their cache to decode the encoded packet. We leverage ideas from cSamp [2] to split the caching responsibilities for interior routers in terms of *hash-range per path per router*. Each interior router is responsible for caching those packets whose header's hash falls in the assigned range for the path. This information is specified by a *caching manifest* produced by the central configuration module. The central

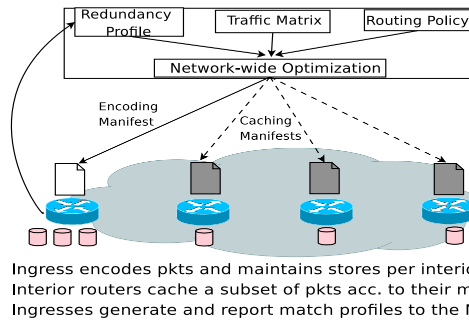
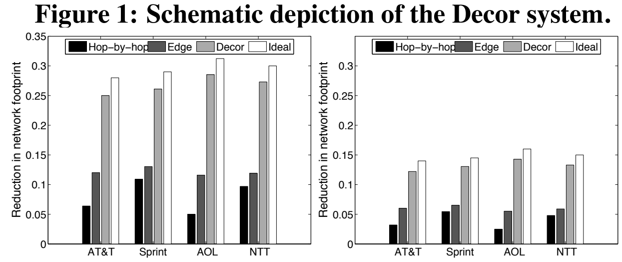


Figure 1: Schematic depiction of the Decor system.



(a) 50% redundancy trace (b) 25 % redundancy trace

Figure 2: Synthetic trace result for 4 ISP topologies

configuration module computes the caching manifests to optimize the ISP objectives e.g. minimizing network utilization, assisting traffic engineering goals. The optimization framework takes into account the prevailing traffic conditions (volume, redundancy patterns), the routing policies and the resource constraints of routers.

We develop lightweight, yet robust mechanisms to ensure system correctness (i.e., every encoded packet can be successfully decoded downstream) in the presence of dynamic cache evictions. Our Click-based prototype can perform duplicate removal at 2.2 Gbps and reconstruction at 8 Gbps.

Evaluation: We evaluate Decor using synthetic and real traces on Rocketfuel topologies. Figure 2 shows the network-wide reduction in footprint over four tier-1 ISP topologies. Decor is $4\times$ better than the naive approach; even the edge-only variant of Decor is 2-3$\times$ better than the naive approach. Further, the difference between Decor and ideal RE (i.e., without any resource constraints) is less than 4%.

References

[1] A. Anand, A. Gupta, A. Akella, S. Seshan, and S. Shenker. Packet Caches on Routers: The Implications of Universal Redundant Traffic Elimination. In *Proc. of ACM SIGCOMM*, 2008.

[2] V. Sekar, M. K. Reiter, W. Willinger, H. Zhang, R. Kompella, and D. G. Andersen. cSamp: A System for Network-Wide Flow Monitoring. In *Proc. of NSDI*, 2008.