On the Optimization of Request Routing for Content Delivery

Walid Benchaita
Alcatel-Lucent Bell Labs France
walid.benchaita@alcatel-lucent.com

Samir Ghamri-Doudane
Alcatel-Lucent Bell Labs France
samir.ghamri-doudane@alcatel-lucent.com

Sébastien Tixeuil
UPMC Sorbonne Universités & IUF
Sebastien.Tixeuil@lip6.fr

ABSTRACT

We present a flexible scheme and an optimization algorithm for request routing in Content Delivery Networks (CDN). Our online approach, which is based on Lyapunov theory, provides a stable quality of service to clients, while improving content delivery delays. It also reduces data transport costs for operators.

1. INTRODUCTION

The role of content delivery networks (CDN) has been constantly evolving since their first deployment to respond to customer needs (from static web content to media streaming), and expectations are still getting higher. It is hence essential to empower content delivery with effective models and solutions in order to deal with the increasing amount of bandwidth-hungry and demanding applications.

Besides, CDN's performance is sensitive to temporarily high request rates or "flash crowds" [6]. Such (potentially unpredicted) huge amount of traffic overloads the delivery infrastructure and causes congestions that lead to service degradation. Request routing (or redirection), which is at the cornerstone of CDN operation, is one of the main keys to improve performance[1]. It has been the focus of several studies and for various purposes [2, 3].

The challenge is to design a request routing solution that is: *optimized* in order to best use the CDN and transport network resources; *feasible online* in order to adapt quickly to the changing conditions; and *flexible* in order to consider the potential objectives set by the CDN operator. To meet these requirements, we use the Lyapunov optimization theory[4] to design an effective online algorithm for request routing, which is then deployed to offer a flexible redirection scheme. Our solution is evaluated through experiments using both synthetic and real workloads. The results show that our multi-objective algorithm provides a stable quality of service to clients, while improving content delivery delays, especially under flash crowd conditions. Moreover, it helps reducing delivery costs by maximizing cache efficiency.

SIGCOMM'15 August 17-21, 2015, London, United Kingdom

ACM ISBN 978-1-4503-3542-3/15/08.

DOI: http://dx.doi.org/10.1145/2785956.2790016

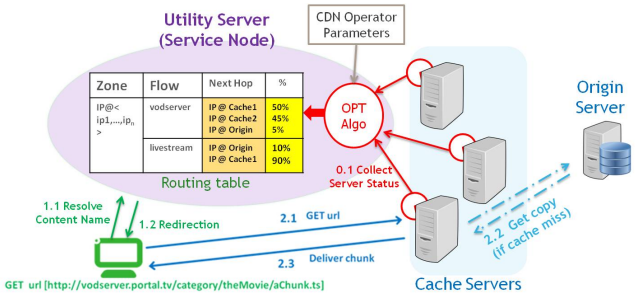

Figure 1: Overview

2. REQUEST ROUTING MODEL AND ALGORITHM

We consider a typical architecture for content delivery as depicted in Figure 1. Initial requests from users are resolved by the CDN service node and a redirection mechanism (DNS-based or HTTP-based, for instance) are used to provide the user host with the relevant cache server to contact. If the content is not available at the cache server, it is retrieved from an upper layer cache or origin server.

In our model, we use a routing table at the CDN service node level to resolve user requests. In our routing table, the clients requests are aggregated according to two criteria: *(i)* user location, and *(ii)* content type. The location plays a key role in reducing the content delivery delay, and is the main parameter used in request routing [5]. The content type characterizes traffic flows, which are classified according to the content provider catalog.

For each input, the routing table provides the most suitable cache servers with specific weights, i.e. the proportion of such requests that each cache server should receive. Our main contribution is the design of an on-line method to update the routing table on runtime (select cache servers and define related weights for each entry). Our multi-objective algorithm, based on Lyapunov optimization theory [4], operates following a slotted time. At the beginning of each time slot, the algorithm uses received inputs in order to compute the new routing table. These inputs are related to both network and cache server status, such as: Average round trip time between clients and cache servers, server loads, hit ratio per flow, etc. Then, the algorithm is composed of two processes.

The first one is a stability process that ensures: *(i)* a continuous service, *(ii)* non-overloaded cache servers, and

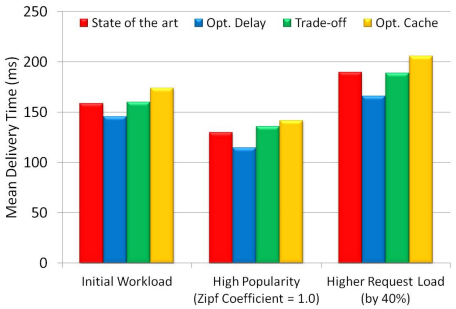

Figure 2: Comparison of average delivery delays using different workloads

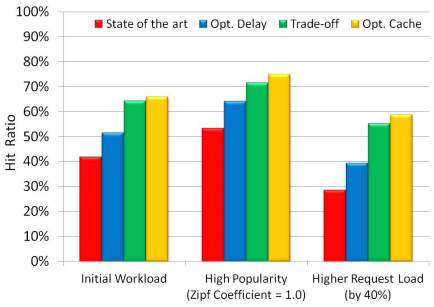

Figure 3: Hit ratio comparison

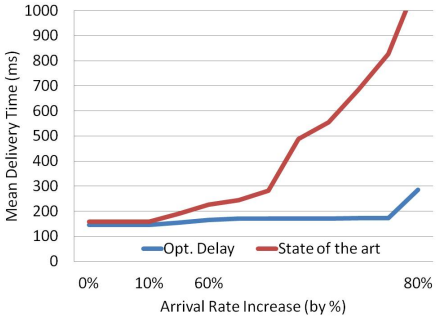

Figure 4: Impact of (increasing) flash crowd rates on delivery delays

(iii) a bounded delivery delay. We use a stochastic model to characterize the system. For every geographical zone i, the clients requests A_i are aggregated according to the type of requested content m into queues: Q_i^m. An allocation to cache servers (the routing table) is noted $U_{i,j}^m$, with j representing each cache. To ensure a continuous service, a relevant allocation will reduce the backlog at all the queues Q. Then, the two other constraints can be represented using virtual queues where, instead of the user requests, the load at each cache server and the average delay of all requests are stabilized. The stability process try to find an opportunistic allocation using the Lyapunov drift minimization techniques[4].

The second process is an optimization one aiming to reduce the delivery costs by increasing the hit ratio at the intermediate cache servers. A high hit ratio (cache efficiency) reduces origin server solicitation and network traffic. The optimization process uses the information related to the hit ratio H_j^m per flow m to direct requests to the cache server j with the highest hit ratio for that flow. The objective function to maximize is: $F = \sum_j \sum_m (H_j^m \times \sum_i (U_{i,j}^m))$

A parameter V is used to characterize the weight (importance) of the optimization process in the allocation scheme. For small values of V, the algorithm privileges the stability process, reducing delivery delays. A high value of V lead to cost optimization, increasing the hit ratio.

3. PERFORMANCE EVALUATION

The performance of our algorithm is evaluated using three targets depicting potential operator objectives: *(i)* optimize delivery delays ($V = 0.15$), *(ii)* optimize cache efficiency ($V = 1.0$), as well as *iii)* a trade-off strategy ($V = 0.55$). A real workload captured from a North American operator CDN was used for evaluations. The network and CDN topologies from this operator are also used. Then, based on the real workload, synthetic ones have been created in order to complete the evaluation by testing alternative conditions. This is done by increasing the content popularity parameter (Zipf's law) and the request load (locally or globally). The obtained results are compared with the best ones from several state of the art solutions that are used in practice.

Figures 2 and 3 assess and compare the delivery efficiency (average delay) and the caching efficiency (hit ratio) for different workloads. The results demonstrate the performance gains brought by our solution, as well as the effectiveness of the multi-objective optimization (performance target).

One of the main advantages of our solution is that it best uses the available resources in order to stabilize the quality of service, even under high request rates. In other terms, it allows reaching the limits of the CDN capacity. Figure 4 demonstrate this by showing the effect of request arrival rate growth on the average delivery delay. Under high pressure, our stability process tends to minimize the miss ratio in order to keep a lower load at all caches and origins. This ensures a good quality of service, while state of the art solutions show very high delivery delays (instability).

4. REFERENCES

[1] A. Barbir, B. Cain, R. Nair, and O. Spatscheck. Known content network (cn) request-routing mechanisms. IETF RFC 3568, 2003.

[2] W. Jiang, S. Ioannidis, L. Massoulié, and F. Picconi. Orchestrating massively distributed cdns. In *ACM International conference on Emerging networking experiments and technologies (CoNEXT)*, 2012.

[3] V. Mathew, R. K. Sitaraman, and P. Shenoy. Energy-aware load balancing in content delivery networks. In *IEEE INFOCOM Conference*, 2012.

[4] M. J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan & Claypool, 2010.

[5] L. Wang, V. Pai, and L. Peterson. The effectiveness of request redirection on cdn robustness. *ACM SIGOPS Operating Systems Review*, 36:345–360, 2002.

[6] P. Wendell and M. J. Freedman. Going viral: flash crowds in an open cdn. In *ACM Internet Measurement Conference (IMC)*, 2011.