On Multi-CDN Delivery Costs Optimization Problem

Yuriy A. Reznik
Brightcove, Inc.
Boston, USA
yreznik@brightcove.com

Guillem Cabrera
Brightcove UK, Ltd.
London, UK
gcabrera@brightcove.com

***Abstract*—This paper studies the problem of minimizing delivery costs in multi-CDN streaming systems. As inputs, this problem receives information about minimum edge traffic volume commitments and edge traffic rates defined in CDN contracts. As input, the problem also receives historical data about CDN traffic volume statistics and predicted volume trends for the remainder of a billing period. As output, it computes load allocation for all CDNs for the remainder of the billing period, minimizing total CDN delivery costs. We show how this problem can be formalized and analytically reduced to a superposition of linear programming and combinatorial search problems. As such, this problem is tractable and practically solvable by existing methods. The paper also discusses the implementation of the proposed methods by multi-CDN systems using the HLS/DASH content steering mechanism.**

Keywords—Streaming, multi-CDN delivery optimizations, linear programming, DASH, HLS, content steering

I. Introduction

As well known, most videos sent over the Internet are delivered using streaming technologies [1-6]. The two most commonly used streaming protocols today are HTTP Live Streaming (HLS) [2] and Dynamic Adaptive Streaming over HTTP (DASH) [3]. Both are international standards. Both use HTTP as the underlying network protocol and employ Content Delivery Networks (CDNs) for distribution [4,5].

However, CDNs have some limits. Some CDNs may not be available in all relevant regions. Some may have a saturated internal network, and some may not have sufficient capacity of edge caches to support the delivery of a vast and diverse catalog of media content. Occasionally, CDNs may also experience outages or technical failures, making them unavailable for some time [4-8].

Considering such limits, large streaming operators increasingly employ multiple CDNs and so-called "CDN switching" technologies as part of their delivery architectures [6-12]. By distributing traffic across multiple CDNs intelligently, such systems can achieve better reliability, scale, and quality of experience (QOS) delivered to end users.

However, using multiple CDNs also comes with extra complexity in managing the costs of the delivery system. The cost models of each CDN can be different and typically include both edge volume rates and commitments on traffic to be routed to each CDN by some dates (commit periods). The problem of managing all such costs and constraints is not entirely trivial to formalize and solve. It is also unclear, for example, to which class of optimization problems it belongs and whether it is solvable by the existing methods. In this paper, we will attempt to answer some of these questions.

Among related prior work, we must mention references [5,7,13-16], discussing various benefits of multi-CDN streaming systems and problems associated with their design. References [14-16] focus on using so-called HLS/DASH "Content steering" features as an essential switching mechanism enabling such design [9-12]. References [14,16] focus on QoE-based optimizations in multi-CDN systems. Reference [13] discusses the cost-driven multi-CDN optimization problem with QOE-based constraints. Reference [13], however, does not consider commit requirements and uses continuous models of CDN cost functions. Compared to the approach in [13], the present paper offers a more holistic and connected-to-practice look at the problem. We look at both commit requirements and discontinuous models of CDN cost functions as essential features of the multi-CDN cost optimization problem.

II. Formalization of the Problem

A. Overall system model

In Fig 1, we show a sketch of a streaming system with K CDNs and a steering server [9-12] employed for directing the traffic. Conceptually, we will think of the steering server as a load balancer, which maintains a particular load distribution:

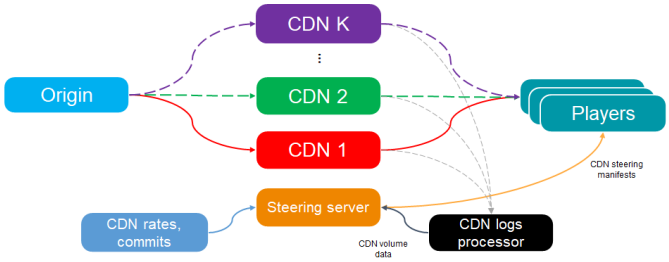

Fig. 1. Model of a multi-CDN system with a content steering server.

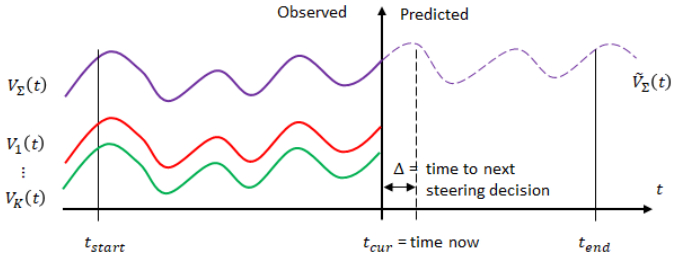

Fig. 2. Conceptual illustration of CDN volume statistics and commit periods.

$$\xi = [\xi_1, .., \xi_K], \ \ \forall i: \ \xi_i \in [0,1], \ \sum_i \xi_i = 1. \qquad (1)$$

and routes requests to each CDN with frequencies approaching load probabilities $\xi_1, \dots, \xi_K$.

The CDN selection or switch decisions may happen at the beginning of each streaming session or at some periodic update interval (TTL), which the system can programmatically define. The CDN load distribution ξ may also be updated at some periodicity, as needed for meeting optimization objectives.

As further shown in Fig 1, as input for its work, the content steering server receives (a) information about CDN costs and commits and (b) CDN edge volume statistics based on data previously delivered by the system.

B. CDN edge volume statistics, CDN volume commits

In Fig. 2, we offer a conceptual illustration of the CDN edge volume statistics. By $V_i(t)$, $i = 1 \dots K$, we denote CDN edge volume quantities reported at each point of time *t*. The combined traffic across all CDNs is denoted by $V_\Sigma(t)$:

$$V_\Sigma(t) = \sum_{i=1}^{k} V_i(t) \qquad (2)$$

Time interval $t \in [t_{start}, t_{end}]$ in Fig. 5 shows the current commit period. The CDN volume commit requirements are defined as follows:

$$V_i^{total} = V_i(\, t \in [t_{start}, t_{end}]) = \int_{t_{start}}^{t_{end}} V_i(t)dt \geq V_i^{commit} \qquad (3)$$

where V_i^{total} denotes the total amount of traffic delivered by CDN i during the commit period, and where V_i^{commit} denotes the minimum volume committed for this CDN.

C. Future traffic. Commit-imposed limits on load factors.

As shown in Fig. 2, at the current time $t = t_{cur}$ we can only observe CDN traffic numbers $V_i(t)$ collected in the past: $t \in [t_{start}, t_{cur}]$. The information about traffic in the remainder of the commit period $t \in (t_{cur}, t_{end}]$ is unknown. But it can be predicted.

The most reliable statistic for such prediction is the combined traffic $V_\Sigma(t)$, as it is not affected by our load distribution decisions between the CDNs made in the past. We will denote predicted future traffic on all CDNs as $\tilde{V}_\Sigma(t)$:

$$\tilde{V}_\Sigma(t) = \text{Pred}(\{V_\Sigma(t), t \leq t_{cur}\}, t), \qquad t > t_{cur}. \qquad (4)$$

Such prediction can be made, e.g., by using a linear predictor [18] or any other classic prediction method operating with a sufficiently long observation window (as traffic statistics, as well known, typically follow periodic patterns at daily, weekly, and yearly cadence).

Once $\tilde{V}_\Sigma(t)$ is computed, the future traffic of each of the individual CDNs $\tilde{V}_i(t)$, can be predicted as follows:

$$\tilde{V}_i(t) = \xi_i^+ \tilde{V}_\Sigma(t), \quad t > t_{cur}, \qquad (5)$$

where $\xi^+ = [\xi_1^+, \dots, \xi_K^+]$ are the CDN load factors assigned for the remainder of the commit period.

Consequently, the total traffic of each CDN during the commit period can be estimated as

$$V_i^{total} \approx \tilde{V}_i^{total} = V_i^- + \xi_i^+ \tilde{V}_\Sigma^+ \qquad (6)$$

where V_i^- is the total traffic delivered since the beginning of the commit period

$$V_i^- = V_i(\, t \in [t_{start}, t_{cur}]) = \int_{t_{start}}^{t_{cur}} V_i(t)dt \qquad (7)$$

and V_Σ^+ is the total predicted traffic on all CDNs till the end of the commit period:

$$V_\Sigma^+ = V_\Sigma(\, t \in [t_{cur}, t_{end}]) = \int_{t_{cur}}^{t_{end}} V_\Sigma(t)dt. \qquad (8)$$

By imposing commit requirements on the estimated total traffic for each CDN: $\tilde{V}_i^{total} \geq V_i^{commit}$, we arrive at the following limits on future load factors ξ^+:

$$\xi_i^+ \ \geq \frac{V_i^{commit} - V_i^-}{\tilde{V}_\Sigma^+}, \ \ i = 1, \dots, K. \qquad (9)$$

The obtained expression (9) defines the limits that the steering server must apply to its effective load balancing factors ξ to ensure that the system satisfies the commit requirements of the CDNs.

D. CDN delivery costs

As discussed in our first example and Fig. 1, CDN edge traffic rates $R_i(V)$ are typically defined as piecewise-constant functions:

$$R_i(V) = \begin{cases} R_i^1 & if \quad V < V^1 \\ R_i^2 & if \quad V^1 \leq V < V^2 \\ \vdots & \vdots \quad\quad \vdots \\ R_i^T & if \quad V^{T-1} \leq V \end{cases}, i = 1, \dots, K \qquad (10)$$

where $V^j, j = 1 \dots T-1$ are the volume thresholds for each pricing tier, and $R_i^j, j = 1 \dots T-1$ are the corresponding CDN rates ($/GB delivered), T is the number of pricing tiers. Parameter V denotes the CDN edge volume (GB) delivered within a billing period. In the simplest case, billing periods may coincide with CDN commit periods.

Next, let us define the total cost of CDN within the current billing period. Recall that the total volume of traffic of each CDN within the current period $t \in [t_{start}, t_{end}]$, can be estimated as (6):

$$\tilde{V}_i^{total} = V_i^- + \xi_i^+ \ \tilde{V}_\Sigma^+$$

where $V_i^- = V_i(\, t \in [t_{start}, t_{cur}])$ denotes traffic already delivered, $\tilde{V}_\Sigma^+ = \tilde{V}_\Sigma(\, t \in (t_{cur}, t_{end}])$ is the predicted traffic for all CDNs, and $\xi^+ = [\xi_1^+, \dots, \xi_K^+]$ is the load allocation between CDNs for the rest of the billing period.

Then, the estimated total cost of operating all CDNs during this billing period (in $) becomes:

$$C_\Sigma(\xi^+) = \sum_{i=1}^{K} \tilde{V}_i^{total}\, R_i\left(\tilde{V}_i^{total}\right) = \sum_{i=1}^{K} \left(V_i^- + \xi_i^+\, \tilde{V}_\Sigma^+\right) R_i\left(V_i^- + \xi_i^+\, \tilde{V}_\Sigma^+\right). \quad (11)$$

E. CDN delivery cost optimization problem

Using the obtained expression for total CDN costs (11), we can now pose the following problem:

Find load factors $\hat{\xi}^+ = \left[\hat{\xi}_1^+, \dots, \hat{\xi}_K^+\right]$, such that:

$$C_\Sigma\left(\hat{\xi}^+\right) = \min_{\substack{\xi^+ \in \{[\xi_1^+, \dots, \xi_K^+]\} \\ 0 \le \xi_i^+ \le 1,\ i=1,\dots,K \\ \xi_1 + \dots + \xi_K = 1 \\ \xi_i^+ \ge \frac{V_i^{commit} - V_i^-}{\tilde{V}_\Sigma^+},\ i=1,\dots,K}} C_\Sigma(\xi^+). \quad (12)$$

The solution is a vector of load factors $\hat{\xi}^+ = \left[\hat{\xi}_1^+, \dots, \hat{\xi}_K^+\right]$ that achieves minimal total edge traffic cost (11) while meeting CDN commit requirements (9).

III. Finding the Solution of the Problem

Observe that in its immediate form, the problem definition (12) does not yet point to any natural technique for solving it. The cost expression (11) in (12) includes discontinuous functions (10), rendering most classic numerical optimization techniques inapplicable.

To gain additional insights, let us next consider a space:

$$J = \left\{[j_1, \dots j_K],\ j_i \in \{1, \dots, T-1\},\ \ i = 1, \dots K\right\}, \quad (13)$$

enumerating all possible combinations of pricing tiers within CDN rate functions (10) that we may be operating in:

$$V^{j_i - 1} \le \tilde{V}_i^{total} < V^{j_i},\ \ i = 1, \dots, K \quad (14)$$

(here, by convention, we assume that $V^0 = 0$ and $V^T = \infty$).

By expanding $\tilde{V}_i^{total} = V_i^- + \xi_i^+\, \tilde{V}_\Sigma^+$, and turning (14) into limits on load values ξ^+ we obtain:

$$\frac{V^{j_i - 1} - V_i^-}{\tilde{V}_\Sigma^+} \le \xi_i^+ < \frac{V^{j_i} - V_i^-}{\tilde{V}_\Sigma^+},\ \ i = 1, \dots, K \quad (15)$$

Observe that now, for each combination of pricing tiers $j \in J$, then we can directly compute the CDN costs (10,11), and impose limits on the applicable load values ξ^+ (15).

By using this observation, the problem (12) turns into:

$$C_\Sigma\left(\hat{\xi}^+\right) = \min_{j \in J} \min_{\substack{\xi^+ \in \{[\xi_1^+, \dots, \xi_K^+]\} \\ 0 \le \xi_i^+ \le 1 \\ \xi_1 + \dots + \xi_K = 1 \\ \frac{V^{j_i - 1} - V_i^-}{\tilde{V}_\Sigma^+} \le \xi_i^+ < \frac{V^{j_i} - V_i^-}{\tilde{V}_\Sigma^+} \\ \xi_i^+ \ge \frac{V_i^{commit} - V_i^-}{\tilde{V}_\Sigma^+}}} \sum_{i=1}^{K} \left(V_i^- + \xi_i^+\, \tilde{V}_\Sigma^+\right) R_i^{j_i} \quad (16)$$

Now, the problem is tractable. The right-side minimization step in (16) is a variant of a linear programming problem. Any classic technique (e.g., the simplex method) is suitable for solving it [12]. The minimization over a space of possible pricing tier combinations $j \in J$ can be implemented by a straightforward combinatorial search. With relatively small numbers of CDNs and pricing tiers in practical systems, this search space is typically small (e.g., 12 for 3 CDNs and 4 pricing tiers). Hence, this problem should be easily solvable.

IV. Additional Practical Considerations

In a multi-CDN system depicted in Figure 1, the above-described logic can reside in the content steering server. Such a server may receive all necessary information about CDN costs, commits, and past volumes as part of its inputs. It may then compute the optimal CDN load allocation load distribution $\left[\hat{\xi}_1^+, \dots, \hat{\xi}_K^+\right]$ periodically, during the billing and commit period, and using such periodic processing to adjust its internal load factors. Such incremental processing and load adjustments should minimize the effects of possible mis-prediction of future traffic and ensure that CDN commits are met by the end of the commit cycle.

Generally, the above-defined problem may produce not a single answer but a space of possible solutions $\left\{\left[\hat{\xi}_1^+, \dots, \hat{\xi}_K^+\right]\right\}$. This space of possible solutions may enable the steering system to perform several additional optimizations (e.g., QOE-based optimizations, etc.) as an extension of this framework.

V. Conclusions

In this paper, we formalized the cost optimization problem in a multi-CDN streaming media delivery system. We have shown this problem maps to a superposition of combinatorial enumeration and linear programming problems, where the combinatorial search space is reasonably small and influenced predominantly by the number of pricing tiers in CDN edge traffic structures. It is practically solvable and can be used to design cost-optimal multi-CDN streaming systems.

References

[1] B. T. Bentaleb, A. C. Begen, C. Timmerer, R. Zimmermann, A Survey on Bitrate Adaptation Schemes for Streaming Media Over HTTP, IEEE Communications Surveys & Tutorials, vol. 21, no. 1, pp. 562-585, 2019.

[2] R.Pantos, and W. May, HTTP live streaming, IETF, RFC 8216, https://tools.ietf.org/html/rfc8216, 2017.

[3] ISO/IEC 23009-1:2022, Information technology - Dynamic adaptive streaming over HTTP (DASH) - Part 1: Media presentation description and segment formats, ISO/IEC, October 2022.

[4] Mind Commerce, CDN Market by Technology, Platform, Application, Service Type, Customer Type, and Industry Verticals 2021 – 2027, 2021.

[5] EBU TR 068, CDN Architectures Demystified, EBU, Geneva, June 2022. https://tech.ebu.ch/publications/tr068

[6] Muvi.com, Multi-CDN switching methods, October 2019. Online: https://www.muvi.com/blogs/multi-cdn-switching-in-streaming-businesses.html

[7] SVTA, SVTA investigation of approaches to CDN delivery, January 2023. https://www.svta.org/2023/01/03/investigating-approaches-to-multi-cdn-delivery

[8] Y. Reznik, G. Cabrera, D. Silhavy, S. Pham, A. Giladi, A. Balk, A. C. Begen, and W. Law, Content Steering: a Standard for Multi-CDN Streaming, ACM Mile-High Video Conference (MHV '24), Denver, CO, February 11-14, 2024.

[9] Apple, HLS Content Steering Specification (v1.2b1), April 2021, https://developer.apple.com/streaming/HLSContentSteeringSpecification.pdf

[10] R. Pantos, RFC 8216bis, HTTP Live Streaming 2nd Edition, Section 7: Content steering, v10, IETF, November 2021. https://datatracker.ietf.org /doc/html/draft-pantos-hls-rfc8216bis#section-7

[11] DASH-IF, DASH-IF Candidate Technical Specification: Content Steering for DASH, Version 0.9.0, DASH-IF, July 2022. https://dashif.org/docs/DASH-IF-CTS-00XX-Content-Steering-Community-Review.pdf

[12] ETSI TS 103 998, DASH-IF: Content Steering for DASH, v1.1.1, ETSI, January 2024, https://www.etsi.org/deliver/etsi_ts/103900_103999/103998/01.01.01_60/ts_103998v010101p.pdf

[13] H.H.Liu, Y. Wang, Y. R. Yang, H. Wang, and C. Tian. "Optimizing cost and performance for content multihoming." Proc. ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM), pp. 371-382. 2012.

[14] D. Silhavy, S. Pham, A. Giladi, A. Balk, A. Begen, and W. Law, Real-Time Streaming Reliability and Performance Optimization Using Content Steering, SMPTE Media Technology Summit, Hollywood, CA, October 16-19, 2023.

[15] Y. Reznik, G. Cabrera, R. Zekarias, B. Zhang, B. Panigrahi, N. Barman, S. Hicks, T. Krofssik, A.Sinclair, and A. Waldron, "Implementing HLS/DASH Content Steering at Scale," Proc. International Broadcast Convention (IBC 2023), September 15-18, 2023

[16] E. Gama, R. Rodrigues-Filho, E. Madeira, R. Immich, and L. F. Bittencourt, Enabling Adaptive Video Streaming via Content Steering on the Edge-Cloud Continuum, IEEE International Conference on Fog and Edge Computing (ICFEC 2024), Philadelphia, PA, USA, May 06-09, 2024.

[17] D. Goldfarb, and M. J. Todd. "Chapter II Linear Programming." Handbooks in operations research and management science, Volume 1, Elsevier, NL, 1989, pp. 73-170.

[18] A. Gersho and R. M. Gray. "Vector quantization and signal compression." Vol. 159. Springer Science & Business Media, 2012.