BLG477E Multimedia Systems
Literature Review About CDNs

Hakan Duran 150200091 - Ahmet Secaettin Alıcı 150190097

June 4, 2025

1 Introduction

Content delivery networks are ground breaking technology but still there is not enough articles and papers in the terms of research. Therefore, what we want to do here, is to prepare a comprehensive literature review on CDNs, offering an in-depth investigation into modern content delivery methods. [1].

We will provide a comprehensive analysis of CDN technologies, including their historical evolution, current status, and future trends. We will identify key performance metrics such as latency reduction, bandwidth savings, and scalability. Our review will also discusses major CDN providers, their service models, and real-world applications. Finally, it highlights existing research gaps and suggests potential directions for future studies on CDN advancements, security enhancements, and cost-efficiency strategies.

2 Little Overview About Content Delivery Networks

With the increasing popularity of the internet, many web services face challenges such as traffic congestion and slow response times due to high demand. To address these problems, Content Delivery Networks (CDNs) are used to improve the performance and scalability of websites. By replicating content and services across multiple mirrored servers in different geographical locations, CDNs ensure that users are directed to the nearest server, which reduces the network load and speeds up response times. CDNs improve network performance by maximizing bandwidth, enhancing accessibility, and maintaining the accuracy of content through replication. They are designed to deliver fast and reliable services by distributing content to edge servers located closer to end-users. These networks consist of

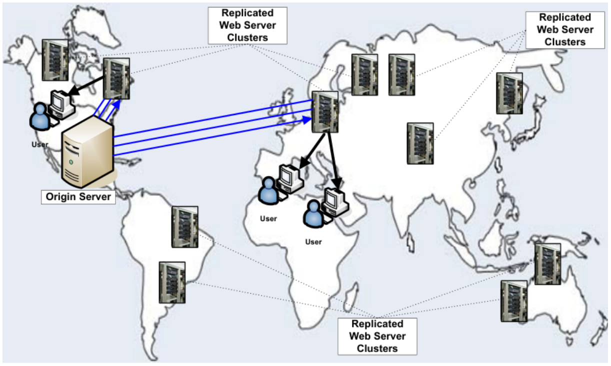
Replicated Web Server Clusters
Replicated Web Server Clusters
User
Origin Server
User
User
Replicated Web Server Clusters

Figure 1: CDN around the world [2]

several key components: the content-delivery infrastructure (edge servers or surrogates), the request-routing infrastructure (which directs users to the nearest server), the distribution infrastructure (which moves content from the origin server to edge servers), and the accounting infrastructure (which tracks usage and generates traffic reports for billing purposes).

The typical content hosted by CDNs includes static content such as images, videos, media files, and advertisements, as well as dynamic web content. The primary customers of CDNs are companies in media, online retail, mobile services, internet service providers (ISPs), and other sectors that need to deliver content reliably and quickly to end-users.

CDNs work by caching content at multiple edge servers, which are strategically placed to handle high traffic volumes and ensure content is delivered quickly. This helps reduce the impact of network congestion and allows for better scalability. Users are directed to the nearest server, thus reducing response time and network congestion. This process is crucial during sudden traffic spikes, such as those caused by flash crowds or viral events, which can overload a website's origin server.

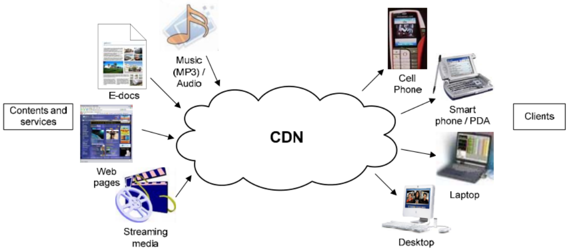

Figure 2: Content of CDNs [2]

Several studies have explored the architecture and functioning of CDNs. These studies highlight the key challenges of building an efficient CDN, such as ensuring content consistency, handling high traffic loads, and optimizing request-routing mechanisms. One of the critical developments in CDN technology is the shift from static content delivery to the ability to handle dynamic and streaming content, such as video-on-demand and live media streaming. Although this technology is still evolving, it has led to the creation of second-generation CDNs focused on improving the delivery of streaming content.

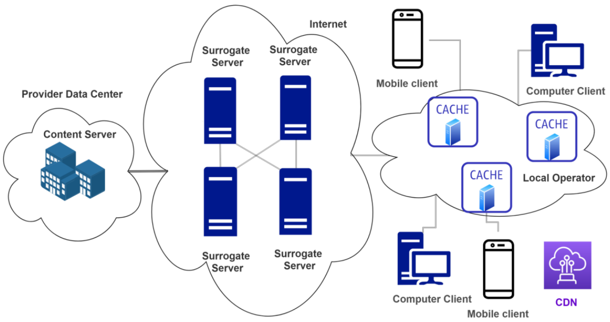

Figure 3: CDN architecture [3]

Over the years, CDNs have become an essential part of the global internet infrastructure. Their use has grown significantly, particularly with the rise of online media and e-commerce. With increasing demands for faster and more reliable content delivery, CDNs continue to evolve, incorporating new technologies and standards developed by organizations like the Internet Engineering Task Force (IETF)[4][5] . As the demand for rich internet content grows, CDNs will remain crucial in ensuring that users experience minimal delays and high-quality service when accessing web content.

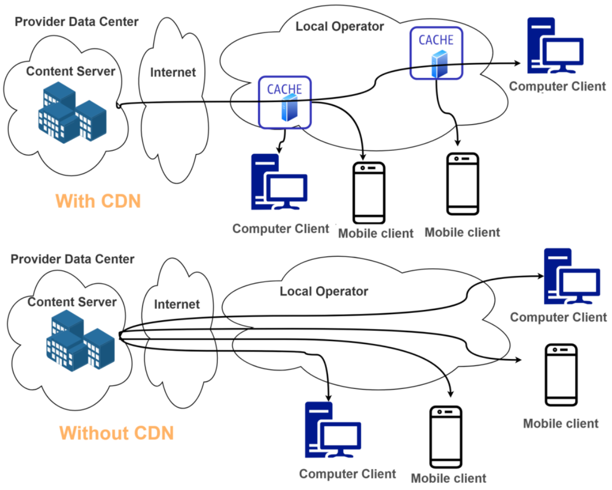

Figure 4: Information flow with CDN and without CDN [3]

3 Request Routing in Content Delivery Networks

A request-routing system helps direct client requests to the best server in a Content Delivery Network (CDN) to deliver content. It uses a set of factors like network proximity, client latency, distance, and server load to choose the most suitable server for the request. In a full-site CDN, the request-routing system directs all requests to the surrogate servers that hold the entire content. In a partial-site CDN, the origin server handles basic content, while the surrogate servers provide additional objects like images or videos.

The request-routing process has two parts: a selection algorithm and a mechanism to inform the client about the server choice. When a client makes a request, the origin server sends the basic content, then redirects the request to the CDN. The CDN's algorithm chooses the best replica server, which delivers the requested content and caches it for future use. This helps ensure fast and efficient content delivery.

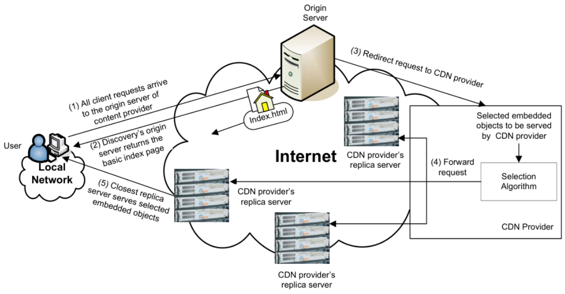
Origin Server
(3) Redirect request to CDN provider
(1) All client requests arrive to the origin server of content provider
Index.html
Selected embedded objects to be served by CDN provider
User
(2) Discovery's origin server returns the basic index page
Internet
CDN provider's replica server
(4) Forward request
Local Network
Selection Algorithm
(5) Closest replica server serves selected embedded objects
CDN provider's replica server
CDN Provider
CDN provider's replica server

Figure 5: Request routing in content delivery networks [2]

3.1 Impact of Request Routing Algorithms on the Performance of CDNs

There are some serious performance problems with the traditional request routing algorithms like Round Robin, Load-based, and Response Time–based routing, which each have limitations. Round Robin and Load algorithms neglect dynamic network conditions, leading to inefficiencies, while Response Time–based algorithms require feedback that might be outdated or incomplete [6].

To address these challenges, algorithm called Worst Surrogate Exclusion (WSE) is proposed. WSE enhances routing decisions by eliminating surrogates with high latency or overloaded links, employing a dual-mode strategy: "Water Filling" for load balancing at lower network utilizations, and "Overloading Prevention" to redistribute traffic before saturation.

WSE's awareness of CDN topology and link speeds enables it to make smarter routing decisions, achieving better performance and resource utilization [6].

3.2 Optimal Content Replication and Request Routing In Content Delivery Networks

A key task in improving CDN performance is content replication and request routing (C2R3), which ensures that content is efficiently delivered from the best possible server. Traditional C2R3 methods often focus on reducing bandwidth usage but tend to overlook content access delay, which is crucial for user experience. To addresses this limitation, C2R3 is framed as an optimization problem aimed at minimizing content access delay in general CDN architectures [7].

Two popularity-based algorithms for content replication and request routing is introduced, contrasting them with the common recency-based methods. The proposed popularity-based algorithms outperform recency-based methods, providing near-optimal performance in real-world scenarios. [7]

3.3 An Interesting Novel Request Routing System Named End-User Mapping

[8] discusses the development and implementation of a new Content Delivery Network (CDN) mapping system called end-user mapping. Traditional CDN mapping systems route content requests based on the location of the local domain name server (LDNS) used by the client. However, this method can be inefficient when the client and LDNS are far apart. The paper introduces end-user mapping, which uses the client's IP address prefix to directly improve the mapping accuracy.

The system's rollout in Akamai's CDN showed significant benefits for clients using public resolvers, especially in countries where the client-LDNS distances were large. Metrics like mapping distance, round-trip time (RTT), time-to-first-byte (TTFB), and content download times all saw improvements, with some metrics showing reductions of up to 50% for the worst-performing clients.

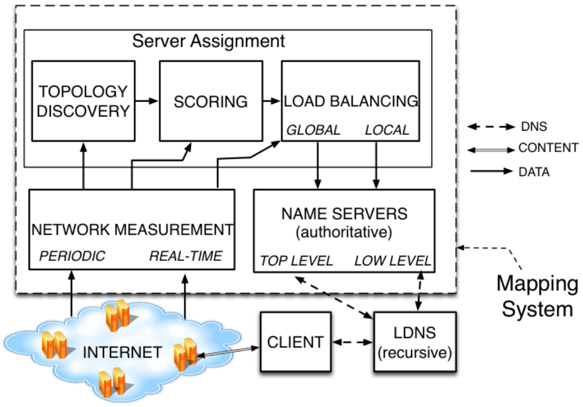

Figure 6: The architecture of the mapping system [8]

3.4 Load Balancing

Load balancing is the practice of distributing traffic among two or more servers. Some load balancing technique utilize a ‘dumb’ load balancing strategy, based on randomizing the distribution of traffic. For example round-robin DNS, a randomized DNS load balancing technique, sends each request to a different server than the last. There are also ‘smart’ load balancing techniques that analyze data in order to decide which is the best server to handle a request. Anycast routing, for example, picks a server based in part on the quickest travel time between the client and the server.

Even before an origin server overloads and stops fulfilling requests, high amounts of traffic to that server can still cause significant latency issues. A GSLB system can distribute that traffic among several different locations, ensuring that no single location is handling so many requests that it causes delay.[9]

3.4.1 A Distributed Control Law for Load Balancing in Content Delivery Networks

[10] introduces a new load-balancing mechanism for Content Delivery Networks (CDNs), aiming to optimize traffic distribution across multiple servers to ensure better system throughput and response times. [10] propose a distributed, time-continuous control law based on fluid flow models, which ensures that server queues are balanced by redistributing excess load. The algorithm was tested through simulations, demonstrating improved performance compared to existing methods, particularly in scenarios with flash crowds, where high traffic demands can cause server overloads.

[10] also compares various dynamic and static load-balancing strategies, such as Random, Round Robin, and Least-Loaded algorithms, showing that the proposed method, called Control-Law Balancing (CLB), performs significantly better in maintaining system balance and response times.

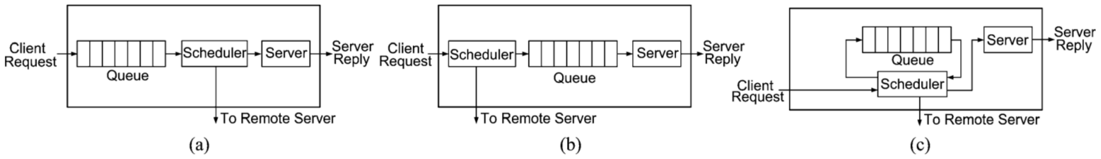

Figure 7: Local load-balancing strategies. (a) Queue-adjustment. (b) Rate-adjustment. (c) Hybrid-adjustment.

For implementation, [10] discuss challenges such as buffering incoming requests and handling redirections, proposing solutions like probabilistic request assignments based on server load differences. Simulations confirm that the CLB algorithm is scalable and effective for large network topologies, even under high load conditions.

4 About Architecture of Content Delivery Networks

4.1 Content Services Network: The Architecture and Protocols

[11] introduces the concept of a Content Services Network (CSN), a layer of network infrastructure that extends the functionality of Content Delivery Networks (CDNs). Unlike

CDNs, which primarily handle content replication and caching, a CSN provides value-added services such as content adaptation, personalization, watermarking, and location-aware data insertion. These services are made accessible to content providers, end-users, ISPs, and CDNs through a set of protocols and infrastructure. The CSN interacts with various network intermediaries, such as application proxy servers and redirection servers, to deliver these services at the network's edge.

A key feature of CSNs is the ability to perform both pre-distribution and post-distribution services. Pre-distribution services involve processing content before distribution, such as video transcoding, while post-distribution services are applied after content has been delivered, adapting the content based on the end user's environment.

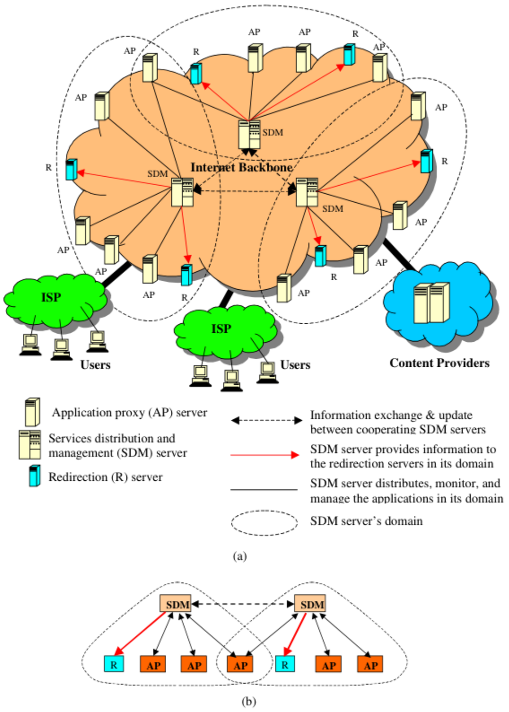
AP
R
SDM
Internet Backbone
ISP
Users
Content Providers
Application proxy (AP) server
Services distribution and management (SDM) server
Redirection (R) server
Information exchange & update between cooperating SDM servers
SDM server provides information to the redirection servers in its domain
SDM server distributes, monitor, and manage the applications in its domain
SDM server's domain
(a)
(b)

Figure 8: The system overview of content services network (CSN)

[10] also proposes the Internet Service Delivery Protocol (ISDP), which facilitates service subscriptions and the interaction between different network elements. [11] concludes by demonstrating the prototype implementation of a video segmentation and keyframe selection system, showcasing CSN's potential to enhance video delivery over the Internet.

4.2 CDNs' Dark Side: Security Problems

There are some vulnerabilities in Content Delivery Networks (CDNs) that affect the communication between CDNs and origin servers, particularly focusing on the back-end connections, which are not visible to end-users [12]. CDNs are essential for enhancing web performance and security, but their back-end security has been largely overlooked. [12] found that many CDNs fail to properly validate the origin server's certificate during HTTPS communication, leaving websites vulnerable to Man-in-the-Middle (MitM) attacks. Moreover, several CDNs support weak cryptographic parameters such as outdated ciphers (e.g., RC4) and weak Diffie-Hellman (DH) parameters, making these connections susceptible to attacks. A framework proposed to test these security issues across 14 major CDNs, identifying over 168,000 websites at risk. There is a lack of transparency for users, as browsers only validate the front-end communication, leaving them unaware of back-end vulnerabilities. To mitigate these risks, CDNs adopt security measures similar to those of modern browsers for back-end communications and ensure that their default configurations align with the latest security standards.

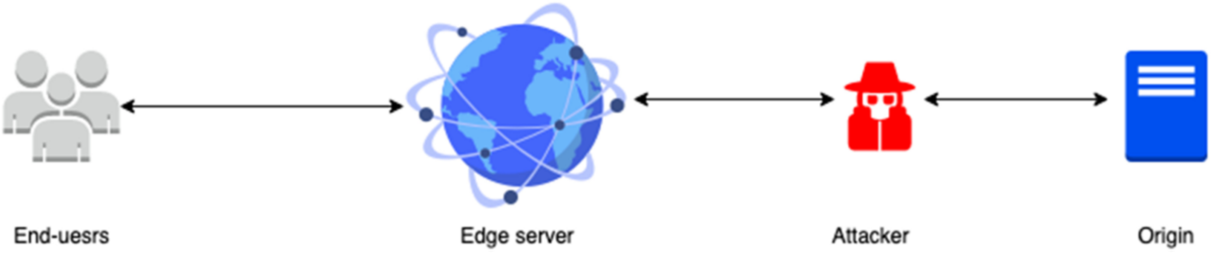

Figure 9: Overview of the attack model. The attacker is an on-path attacker in the back-end communication.

5 Some Strategies For Streaming in CDNs

5.1 Adaptive Bitrate Streaming

5.1.1 Enhancing Content Delivery Networks: Optimizing Performance and Quality of Experience

In [13], Bhat et al. introduce a novel architecture for adaptive bitrate (ABR) streaming, termed SABR, which leverages Software-Defined Networking (SDN) to enhance the performance of Content Delivery Networks (CDNs) in video streaming applications. As video streaming continues to dominate internet traffic, with a significant share attributed to video-on-demand (VoD) and live streaming services, optimizing the delivery of video content becomes increasingly important. Traditional methods of video streaming often face challenges related to bandwidth variability, quality fluctuations, and server overload, particularly under high traffic conditions. SABR aims to address these issues by integrating SDN with ABR streaming protocols.

The key idea behind SABR is to assist video clients in making more informed decisions about which video segments to retrieve and from which cache, based on real-time network information. While traditional ABR streaming systems rely on the client's estimation of available bandwidth and segment quality, SABR uses SDN to provide dynamic feedback regarding the network's bandwidth availability, cache occupancy, and the quality of video segments hosted on various caches across the network. This enables the client to optimize video segment retrieval, improving the overall Quality of Experience (QoE) for viewers. SABR's design ensures that clients retain full control of their streaming algorithms, reducing the burden on network control planes while still benefiting from the network assistance.

SABR is implemented and evaluated in a real-world SDN-enabled testbed, CloudLab, using the Dynamic Adaptive Streaming over HTTP (DASH) protocol. The evaluation demonstrates that SABR significantly improves key performance metrics such as cache hit rates, playback bitrate, and rebuffering ratio compared to baseline ABR systems. By providing the client with real-time bandwidth estimates and cache maps, SABR reduces server load by offloading content delivery to caches closer to the client. This network-assisted streaming model also enhances the client's ability to handle bandwidth fluctuations and avoid quality degradation due to network congestion or delays.

The exploration of different content placement strategies for SABR is very important. Local caching, global caching, and quality-based caching are evaluated to determine how

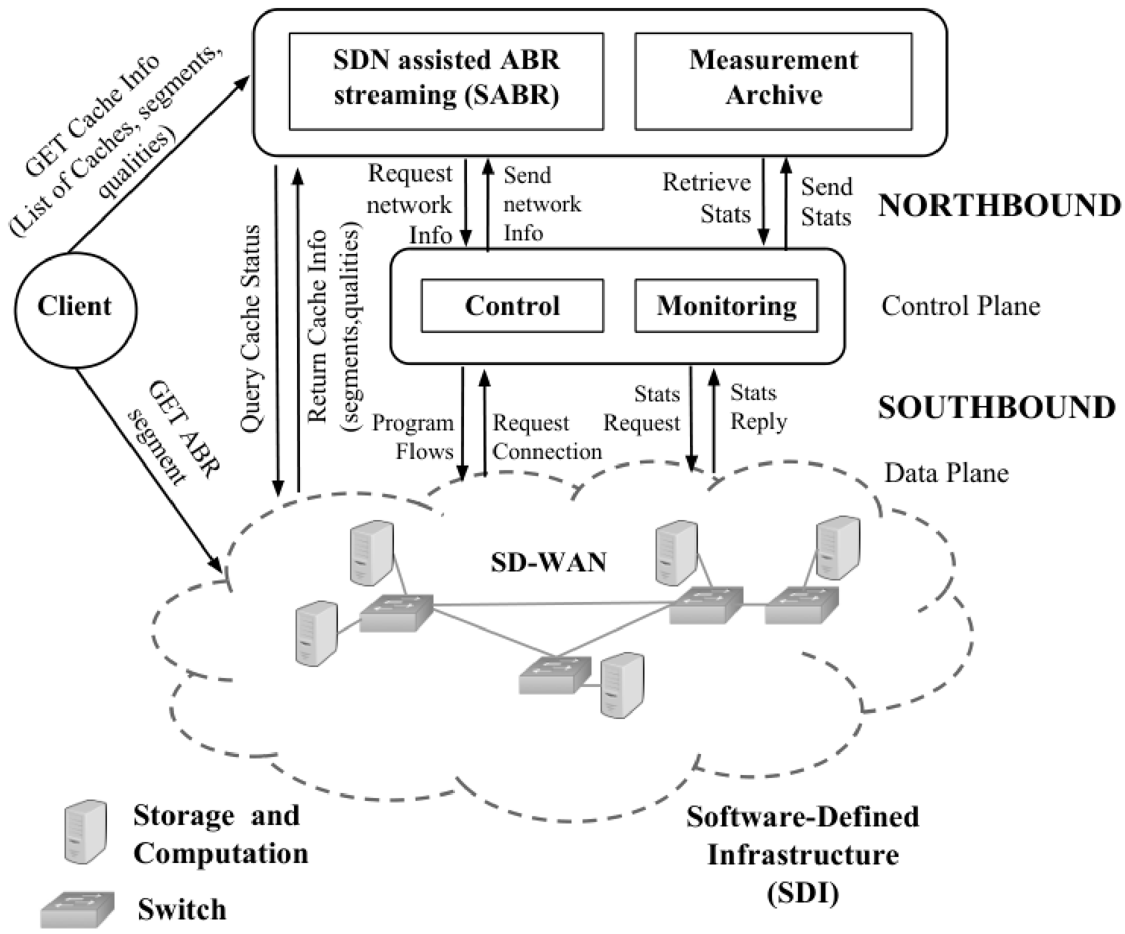

Figure 10: SDN assisted adaptive bitrate video streaming (SABR) architecture

these strategies impact the overall QoE and system performance. The results show that SABR performs best when combined with a global caching strategy, where the cache network is aware of content availability across the system. This approach improves cache hit rates and network utilization, further enhancing QoE for the users.

[13] contributions lie in its design of the SABR architecture, its formalization of the ABR problem in the context of network-assisted streaming, and its implementation and evaluation through real-world experiments. By optimizing both the content delivery and the caching architecture using SDN capabilities, SABR offers substantial improvements in video streaming performance. It enables a more flexible and scalable approach to video delivery, especially in environments where high-quality streaming is essential. This work paves the way for future advancements in SDN-assisted content distribution systems, and its techniques could be extended to other ABR protocols beyond DASH, such as HLS (HTTP Live Streaming) and HDS (HTTP Dynamic Streaming), which share similar requirements.

In conclusion, the potential of SDN in transforming the landscape of adaptive bitrate streaming are demonstrated by providing enhanced network intelligence that allows clients to make better streaming decisions. SABR's integration of SDN-assisted features, such as real-time network monitoring and dynamic path selection, makes it a promising solution for optimizing content delivery in modern CDNs.

5.1.2 Enabling adaptive bitrate algorithms in hybrid CDN/P2P networks

[14] by Yousef et al. discusses the challenges of implementing adaptive bitrate (ABR) algorithms in hybrid content delivery networks (CDN) and peer-to-peer (P2P) systems. As video traffic has become a dominant portion of global internet traffic, the paper focuses on how ABR algorithms, essential for optimizing video quality over varying network conditions, can be adapted for P2P environments where bandwidth is more dynamic. The study introduces two new methodologies—BufDel and NetDel—to enhance ABR efficiency in these hybrid networks.

ABR algorithms have revolutionized video streaming, enabling quality optimization by adapting to network conditions such as available bandwidth and buffer levels. However, ABR algorithms were primarily designed for CDN networks, and their performance in P2P systems, where peer-to-peer segment sharing occurs, has not been well studied. The paper identifies the need for improvements in ABR algorithms for hybrid CDN/P2P networks to maintain good quality of experience (QoE).

ABR algorithms are categorized into buffer-based, throughput-based, and hybrid approaches. Impact of P2P network conditions, such as peer heterogeneity and varying bandwidth, on ABR decisions, are too much which can lead to inefficiencies like quality oscillations or frequent re-buffering. These issues are exacerbated when segments are fetched from P2P caches, often leading ABR algorithms to make suboptimal bitrate selections.

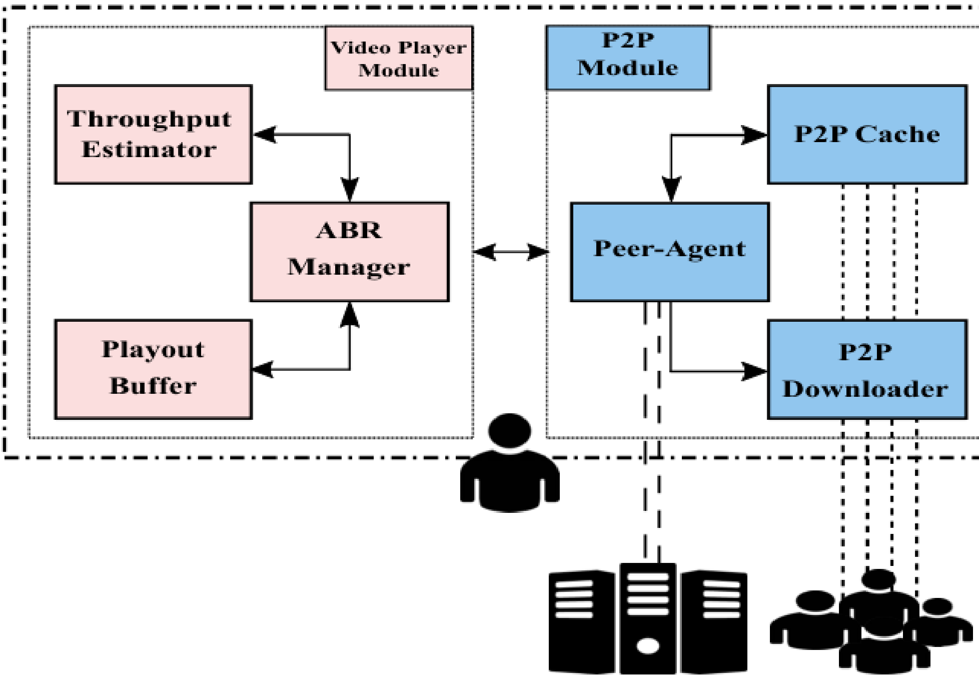

Figure 11: Hybrid CDN/P2P HTTP adaptive streaming

To address these issues, [14] proposes the Response-Delay methodology, which introduces a delay in delivering segments from the P2P cache to align ABR decisions with actual bandwidth availability. This delay modulates the download time perceived by the ABR, making it appear as if segments were fetched from a CDN, preventing ABR from overestimating bandwidth and selecting overly high bitrates.

Two delay models—BufDel and NetDel—are introduced. BufDel uses the buffer level to adjust the delay based on the current buffer occupancy, while NetDel adjusts based on the highest available bandwidth between CDN and P2P. These approaches help mitigate issues like buffer underestimation or overestimation of available bandwidth, thus improving the stability and efficiency of video delivery in hybrid systems.

The experimental evaluation, using MATLAB simulations and real-world 3G bandwidth traces, demonstrates that the proposed methodologies significantly improve both QoE and P2P efficiency. The authors show that NetDel is particularly effective at stabilizing streaming quality and increasing P2P resource utilization, while BufDel increases the average quality and reduces the frequency of quality switches.

Lastly, [14] includes commercial service trials conducted with real-world streaming platforms, validating the effectiveness of BufDel and NetDel in real scenarios with up to 15,000 concurrent peers. These trials confirm that NetDel provides better P2P efficiency and lower rebuffering duration compared to BufDel, with the trade-off being a slightly higher number of rebuffering events. The authors conclude by suggesting that NetDel is the preferred approach for cost-efficient, stable, and smooth streaming, while BufDel is more suitable for cases where higher average quality is prioritized.

6 Conclusion

CDNs have changed how web content is delivered, making it faster and more efficient. As technology grows, CDNs will continue to improve and expand. Future challenges include security, cost reduction, and better content adaptation. Our literature review covered and examined some of those, their advantages and disadvantages and many more.

References

[1] A. Vakali and G. Pallis, "Content delivery networks: Status and trends," *IEEE Internet Computing*, vol. 7, no. 6, pp. 68–74, 2003.

[2] A. M. K. Pathan and R. Buyya, "A taxonomy and survey of content delivery networks," Grid computing and distributed systems laboratory, University of Melbourne, Technical Report 4, 2007.

[3] B. Zolfaghari, G. Srivastava, S. Roy, H. R. Nemati, F. Afghah, T. Koshiba, A. Razi, K. Bibak, P. Mitra, and B. K. Rai, "Content delivery networks: State of the art, trends, and future roadmap," *Journal of Computer Networks and Communications*, vol. 2021, pp. 1–18, 2021, published online in 2021.

[4] M. Day, B. Cain, G. Tomlinson, and P. Rzewski, "A model for content internetworking (cdi)," RFC 3466, February 2003, informational.

[5] A. Barbir, R. Penno, R. Chen, M. Hofmann, and H. Orman, "Network access protection (nap) architecture," RFC 3835, August 2004, informational.

[6] M. Masa and E. Parravicini, "Impact of request routing algorithms on the delivery performance of content delivery networks," *IEEE International Conference on Communications, ICC*, pp. 5–12, 2003, presented at the 2003 IEEE International Conference on Communications. [Online]. Available: https://ieeexplore.ieee.org/document/1204449

[7] P. Amani, S. Bastani, and B. Landfeldt, "Towards optimal content replication and request routing in content delivery networks," *IEEE ICC 2015 - Next Generation*

Networking Symposium, pp. 5733–5739, 2015, presented at IEEE ICC 2015. [Online]. Available: https://ieeexplore.ieee.org/document/7403796

[8] F. Chen, R. K. Sitaraman, and M. Torres, "End-user mapping: Next generation request routing for content delivery," *SIGCOMM '15*, pp. 167–180, 2015.

[9] Cloudflare, "Global server load balancing (gslb)," 2023, accessed: 2023-06-04. [Online]. Available: https://www.cloudflare.com/learning/cdn/glossary/global-server-load-balancing-gslb/

[10] S. Manfredi, F. Oliviero, and S. P. Romano, "A distributed control law for load balancing in content delivery networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 1, pp. 55–68, 2013.

[11] W.-Y. Ma, B. Shen, and J. Brassil, "Content services network: The architecture and protocols," *IEEE Internet Computing*, vol. 5, no. 2, pp. 32–39, 2001.

[12] B. Shobiri, M. Mannan, and A. Youssef, "Cdns' dark side: Security problems in cdn-to-origin connections," *Digital Threats: Research and Practice*, vol. 4, no. 1, p. 3, 2023.

[13] D. Bhat, A. Rizk, M. Zink, and R. Steinmetz, "Network assisted content distribution for adaptive bitrate video streaming," in *Proceedings of MMSys'17*. Taipei, Taiwan: ACM, 2017, pp. 1–14, accessed: 2023-06-04.

[14] H. Yousef, J. Le Feuvre, P.-L. Ageneau, and A. Storelli, "Enabling adaptive bitrate algorithms in hybrid cdn/p2p networks," in *11th ACM Multimedia Systems Conference (MMSys'20)*. Istanbul, Turkey: ACM, 2020, pp. 1–12.