Video Content Delivery Enhancement in CDNs based on Users' Social Information

Praboda Rajapaksha[†‡§], Reza Farahbakhsh[†], Samin Mohammadi[†], Matthew N. Dailey[‡], Noël Crespi[†]
[†]Institut Mines-Télécom, Télécom SudParis, CNRS UMR 5157 SAMOVAR, France,
{reza.farahbakhsh, samin.mohammadi, noel.crespi}@it-sudparis.eu
[‡]Asian Institute of Technology, Thailand, mdailey@ait.asia
[§]Uva Wellassa University of Sri Lanka, praboda.rajapaksha@telecom-sudparis.eu

***Abstract*—With the huge growth of multimedia communication and digital content availability, energy efficient content delivery became an important research topic with the goal of reducing energy consumption of the intermediary nodes while providing better services and QoE to the end users. In this paper we focus on the subject of reducing the overall network energy consumption in accessing user generated content over social media platforms. We propose an approach, namely *SocialiVideo* which enables users to directly share their generated video content among existing social connections. We combine the approaches used in CDNs and P2P networks together with social connections between people in order to shorten the path the data traverses on average, and improve the latency. *SocialiVideo* places video content in users' premises (e.g., set-top-boxes) and serve others using a P2P connection. To this end, we use users' geolocation information retrieved from their network data (IP address) as well as subscribed social networks (e.g. current city attribute in Facebook) and social characteristics (e.g. friends list, and activities, etc.). In-order to evaluate the performance of the proposed solution, we implement our prototype based on Facebook/Akamai content delivery approach and evaluates the performance with reference to the current solution of the Facebook. Based on the results, *SocialiVideo* unload the traffic of the network and CDN, thus reduces network energy consumption and provides advantages for multiple entities, including CDNs and ISPs, as well as better QoE for end users.**

I. Introduction

With the eruption of Online Social Networks (OSNs), video traffic and especially user generated videos have quickly become the dominant fraction of Internet data traffic. At present, the second and third most traffic generating websites are Facebook and YouTube respectively[1], and YouTube is considered as the largest video streaming platform followed by Facebook[2]. These evaluations are becoming more relevant with the increase of streaming traffic demands generated via OSNs. Therefore, many of these content providers rely on the 3rd party Content Delivery Networks (CDNs) which provides advanced content delivery services (e.g., Facebook uses Akamai services).Usually these content providers are very efficient at serving popular content by using locality and temporal patterns. However, a large portion of the content is user generated, and delivering this mostly unpopular content consumes more energy and is a complex task due to the lack of content availability and the data sparsity. That means, user request needs to be forwarded to the content server, which is costly for both end users (i.e., long access delays) and content provider (i.e., proper resource allocation). This may not be an issue for a single or few queries, but rather for large content providers like YouTube or Facebook that may generate millions of queries for only a few visits of different videos.

A potential solution to alleviate this problem would be to use end users' available premises such as their set-top-box (STB), which are connected to the Internet and has storage to cache the content in-order to provide peer-assisted content delivery services. There are some efforts in literature with the same aim such as a study [1] that studied the possibility of offloading 60% to 80% of the total traffic using peers uplink. However, the majority of these mechanisms use swarming protocols to cache the content partially or completely.

On the other hand, due to huge involvement of users in generating content in the last decade, energy consumption of ICT devices has increased significantly. Authors in paper [2] note that the world's ICT systems consume about 1500TWh of electricity annually and it is approaching 10% of the world's electricity production. Hence, many studies focus on reducing energy consumption of the networking elements in the Internet, including a number of greening concepts. Several other studies pays particular attention to energy efficient methods in data-centers and CDNs such as (i) turning off servers during low traffic periods [3] without violating load capacities (ii) placing replicated content servers closer to end users in order to reduce the distance (hop count) to the origin server. Comparably, energy consumption is increasing due to growing link/traffic rate [4], hence, video content dissemination over the Internet could generate more traffic and consume more energy, particularly for packet processing, switching, and storage mechanisms [5].

In this paper we propose a social peer-assisted content delivery approach, *SocialiVideo* which aims to deliver user generated videos efficiently based on the users' social information obtained from OSNs and network layer such as location and network information. *SocialiVideo* alleviates energy consumption of the network and increases QoE. It allow other users who belong to her social graph and are located in the same location (can be considered same city, country, etc.) of the content owner to access them directly using a P2P communication. To deliver a content, *SocialiVideo* provides

[1]http://www.alexa.com/topsites
[2]https://www.comscore.com

two possibilities; (i) if the content requester and owner are in the same network, video can be streamed in offline mode for available set of videos in the users' local resources. (ii) it also can provide the possibility to stream videos from the usual path through CDN edge servers when local video is unreachable. Our performance evaluation shows that, introduce content delivery mechanism can reduce substantially the load of traffic and processing of the network as well as in the social network CDNs infrastructure.

The rest of the paper is organized as follows. Section II provides a literature overview on the topics related to this study. Next, Section III introduces *SocialiVideo* approach and compare it with the existing approach in Facebook. Finally we evaluate the performance of SocialiVideo in section IV and section V concludes the paper.

II. Literature Review

We study related previous work based on following three domains:

Content delivery enhancement in CDNs: Recently, researchers show that, hybrid architectures based on both CDNs and P2P for large scale video distribution over the Internet is an efficient approach [6] [7]. These hybrid CDN-P2P systems combine best of both CDNs and P2P and user content can be delivered using CDN edge server, P2P, or both networks [8]. Another important effort, CCN was introduced by Jacobson et al. [9] which can be consider as a major turning point of content delivery. Several studies have been done on CCNin order to investigate optimal energy consumption in CCN networks based on static and dynamic in-network caching policy: PPCSA[3] designed for video content delivery [10].

Utilizing social networks and users information in CDNs: To the best of the authors' knowledge except few studies, the idea of using social features of users information available in OSNs to enhance CDN performances have not been studies well enough in the literature. Nevertheless, authors in [11] proposed an OSN assisted caching strategy to improve performance of CDNs. Furthermore, geographical feature analysis for video content dissemination and streaming has been deliberated in many studies. Few examples are; examining geographical locality of IPTV [12], corellation between content and geographical locality of YouTube videos [13]. Interestingly, few studies [14], [15] elaborated that users likely to communicate with one another in the same geographical region. Hence, this prove that locally generated content is more popular among local community and our solution can be use to disseminate those content by considering CDN as a black-box.

Cashing mechanisms and P2P video content delivery mechanisms:

The mehanism that proactively place the content in the caches of peers (e.g., in home gateways, set-top-boxes), called nano data centers for P2P VoD architectures has been studied by Nikolaos L. et al. [16]. In another study [17], focused on a P2P decentralized model for VoD streaming assuming that only one server has all the chunks. Another study proposes a Push-to-Peer system [18] for STBs in which a video is first pushed (e.g., from a content creator) to a population of peers to store and work similar to traditional P2P system. SocialStreaming [19] is another algorithm designed for P2P video streaming. It fragments videos and cash them in different peers and when another peer request this video, requested peer join with SocialStreaming system and obtain video fractions/full. Even though the objective of this study and our system is similar, our approach has several differences such as there is no prior selection of neighbors, peer contact is based on the user's location, and video delivery is on-demand. Finally, Akamai NetSession[4] [1] is another similar approach which runs in the users' browser enabling them to perform P2P transfers with other neighboring users. This approach consume end users' device CPU and disk space to store the content. NetSession has a control plane that keeps track of the content availability of each peer where each peer can store fixed size pieces of the content with the hashes separately.

Considering all that has been mentioned so far, to the best of the author's knowledge *SocialiVideo* is a novel approach which can advance the state of the art on the problem of enhancing user generated content delivery.

III. System Architecture

This section summarizes how existing Facebook content delivery is working and next provides detail on how the proposed solution is able to enhance the performance.

A. Current video delivery solution in Facebook

Facebook uses Akamai content delivery services which reduces 2.5 times delay in the delivery than accessing Facebook native servers [20]. As illustrated in Figure 1, Akamai operates using a mapping system for directing requests to CDN servers. A CDN mapping system enables a peer to locate another peer in the network after associating with the CDN. A peer may be a personal computer, a server or any other computing device that has sufficient computing and bandwidth capabilities to support some of the edge server functionality. One widely used peer category is personal computers, allowed to join a swarm to generate a decentralized P2P network in-order to be a part of the CDN infrastructure to provide resources and data. Using this hybrid approach, Akamai customers' content may be delivered from the CDN edge network, from the P2P network, or even from both of these networks.

When users upload content to the Facebook, it will store in the closest edge network. The closest edge server is chosen based on a number of factors (e.g. real-time loss and latency, real-time capacity and demand information, and different traffic categories) by the mapping system depending on the historical and current data (e.g., ping, traceroute, BGP data, logs). Consider a user (e.g., U1 in Figure 1) uploaded content is in the edge network. Next, a CDN edge network is used to prime the P2P network, which may be used to take over some

[3] https://www.comscore.com

[4] http://www.akamai.com/client/

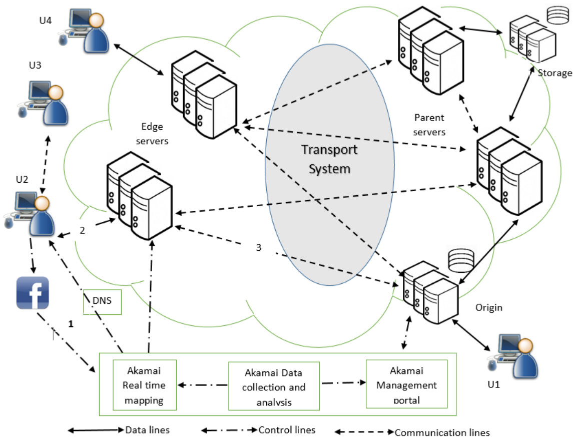


Fig. 1. Facebook/Akamai video content delivery (FB/AK case)

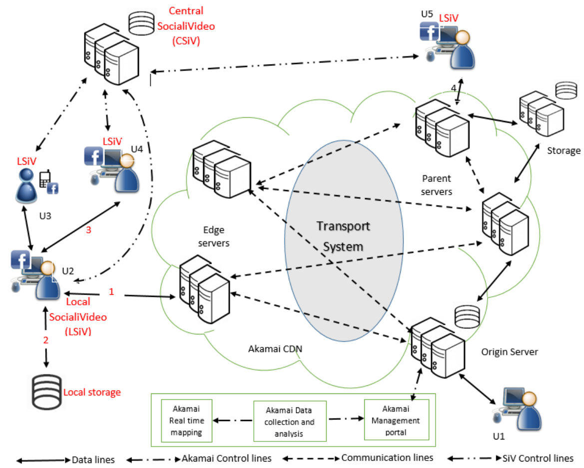


Fig. 2. SocialiVideo system architecture (SiV case)

of the content delivery requirements when U2 requests U1's uploaded content. The decision of whether to use edge network or peer network is based on the present network conditions.

After directing a user's request to a chosen edge server/peer (Arrow 2), if the content is available in that edge node, U2 receives content directly as an HTTP flow. Otherwise, the edge server will contact the origin server (Arrow 3) and pull the content via a transport system. Then, the Akamai server will delivers content to U2 as an HTTP stream.

B. SocialiVideo (SiV) solution

In *SocialiVideo* approach (Figure 2), there are one or more central SocialiVideo severs (CSiV) which help peers to locate their desired resources. CSiV is a central server that keeps track of the user profile information (location, social relationships, etc.) and video metadata (video URL-local storage and Akamai edge server, video name, etc). Peer sends a request to CSiV to determine the address of the local peer that has the requested content (i.e. video). In the proposed system, P2P overlay is generated using users' social information such as location, network details, and Facebook friendship relationships. To be a part of the system, users execute P2P-SiV Web application on their device and authenticate using Facebook credentials. Obtaining a social graph of a user from the Facebook graph API is important to identify the content privacy. Local SocialiVideo (LSiV) is another element of our solution which executes as a daemon process in the users' premises or local storage allowing to download new videos or delete the existing ones. LSiV acts as a system's data plane and therefore locally available content and information is synchronized between LSiV and CSiV according to the available updates.

Figure 2 illustrates the functional view and data traffic flow of the *SocialiVideo*. This solution is based on three implemented algorithms. Algorithm 1 executes when a user uploads/shares a video in the Facebook while SiV application is executing on her device. When another user tries to stream videos shared by her friends, Algorithm 2 evaluates the possibilities by accessing CSiV and finally, Algorithm 3 performs cache management for LSiV.

Algorithm 1 uploading/sharing a video

procedure SIV–PROCEDURE-1
    for each peer i **do**
        if LSiV of i is active **then**
            if Ui share V1 into the FB **then**
                stream from CDN;
                fetch V1's video link from the FB
                download V1 into Ui's local storage
                update CSiV/LSiV
            else if Ui upload V2 into the FB/SiV **then**
                upload into the CDN edge server
                copy V2 into Ui's separate local storage
                update CSiV/LSiV
            end if
        end if
    end for
end procedure

1) URL detection strategy: SiV can fetch both shared (YouTube) and users' uploaded video links from a users' Facebook wall. This procedure is illustrated in Algorithm 1 by using two cases refer to Figure 2.

Case-1: U2 shares the link of a video (V1) which is already available in the Internet (e.g., YouTube or user-uploaded video in the Facebook). Then, as shown in Arrow 1 of Figure 2, CDN streams and delivers V1 to U2. At the same time, SiV fetches V1's link and downloads the video into U2's local storage (Arrow 2) which later will acts as a server to stream it to nearby users. Finally, CSiV and LSiV entries will be updated and synchronized according to the video's metadata and users' social information.

Case-2: U2 uploads his own video-V2 into the Facebook/SiV and then, SiV automatically copies it into the related local storage. After that, SiV extracts V2's link from U2's Facebook wall and synchronizes CSiV and LSiV entries. This process allows user to upload videos to SiV application which then permits other users to stream from her local storage.

Algorithm 2 streaming a video

```
procedure SiV–PROCEDURE-2
    for each peer i and j do
        if LSiV of i and j are active then
            if Uj view or share V1/V2 in FB then
                if CSiV has Ui-Uj FB friendship relationship & CSiV has same location for Ui and Uj then
                    Uj stream from Ui's local storage
                    if stream from Ui is failed then
                        Uj stream from CDN
                    end if
                else
                    Uj stream from CDN
                end if
            end if
        end if
    end for
end procedure
```

Algorithm 3 local cache management

```
procedure SiV–PROCEDURE-3
    for each peer i do
        if SocialiVideo of i is active then
            if Ui delete V1/V2 link from FB then
                remove V1/V2 from Ui's local storage
                update CSiV/LSiV
            end if
            if Ui's local memory exceeded then
                remove unpopular oldest video from the Ui's local cache
                update CSiV/LSiV
            end if
        end if
    end for
end procedure
```

2) Neighbor selection strategy: Algorithm 2 elaborates how *SocialiVideo* allows other users to stream from a local storage. One of the main social attributes used in SiV is users' current location (city, country, or network). If a user needs to stream her friend's video, the SiV application executed in her device checks CSiV whether they are in the same location. If their location is same, she can stream the video from friend's local server or CDN edge server based on the content availability of the local server and also current network conditions. As shown in Figure 2, if U2, U3, and U4 are in the same physical location and if the video is available in U2's LSiV and U2-U3 and U2-U4 have Facebook friendship relationships, then U3 and U4 can stream the video from U2's LSiV (Arrow 3 in Figure 2). Additionally, if any transmission error occurs when try to stream from U2's LSiV, CSiV is capable of redirecting the communication back to an edge server (Arrow 4). By default, an end user cannot determine from which server the video is streaming. The SiV application chooses the best streaming server based on present traffic conditions (i.e., total response time) of the network. If U2, U3, and U4 are in different locations, then streaming will be performed via Akamai CDN.

3) Local memory management: To have an efficient memory management in the local devices (e.g. : devices with high capacity and better performance such as STBs), we considered some policies and added some attributes in the LSiV. Firstly, LSiV only allows one copy of the same video to be cached. If users re-share the same link, LSiV detects that the video is already available by comparing to the list of downloaded videos based on their identification number/name. Secondly, when a user deletes a video links from her Facebook wall page, LSiV automatically removes the video from the local server and synchronizes CSiV and LSiV based on that entry. Algorithm 3 shows that the process of optimizing the local storage space. Thirdly, LSiV keeps track of most popular videos based on the rank obtained from number of views and the published date. This helps to open some space in the local storage when memory exceeded and there is a need for space to download new videos. Lastly, if one video has higher ratings than other videos, LSiV keeps the video copy for longer time than other videos. Therefore, when LSiV needs to remove some videos from the local storage, as shown in Algorithm 3, it consider the mentioned metrics and remove oldest unpopular videos.

IV. Analysis of SocialiVideo

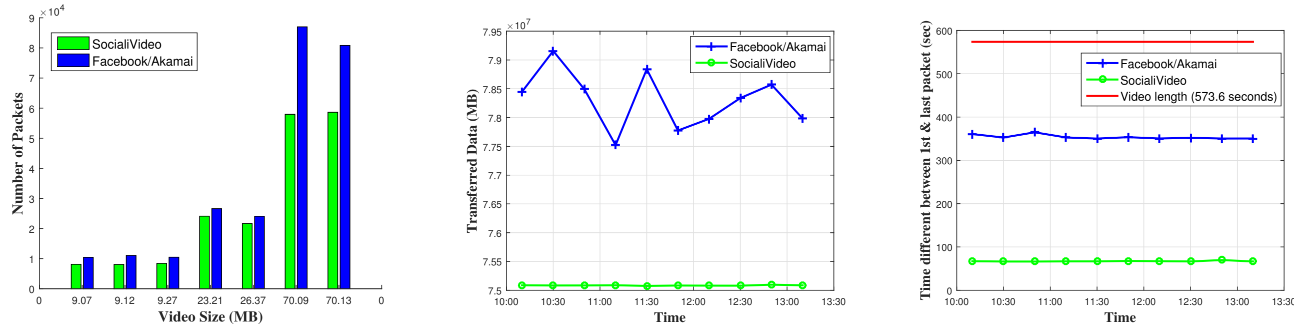
This section evaluates the performance of *SocialiVideo* and compares it to the current Facebook/Akamai (FB/AK) video delivery approach.

A. General comparison

This part differentiates few main characteristics of SiV with the FB/AK approach. Our solution architecture is based on the FB/AK approach, but SiV uses P2P distributed services. Session path and number of hops from the end users also important, which is higher in the FB/AK due to forwarding queries to the edge server clusters placed in ISPs/PoPs. In addition, SiV solution works as an ISP friendly service (in cases that users are in the same network/city) as the content is in users' local server and also, it reduces inter ISP traffic due to P2P streaming. Another point is capital intensive, which in SiV is much lower as it uses existing user premises and the existing underline network. However, implementing CDN replica server clusters in different places is very costly. Our presented analysis in following sections show that SiV solution is reducing the global propagation delay, which is roughly 100 ms for a one way interactions in a usual session [21]. Lastly, both FB/AK and SiV use HTTP flows to deliver streaming content as a reliable transmission. More information about the energy, latency, and traffic parameters is given in the following sections.

B. Performance evaluation

The SiV solution aims to reduce power consumption of network devices while minimizing the traffic load on the



(a) Number of delivered packets (b) Transferred traffic (#Bytes) (c) Packet arrival time

Fig. 3. Performance evaluation of video content delivery of SocialiVideo vs. Facebook/Akamai

intermediate devices by serving video content in a P2P fashion between closer users. To evaluate the enhancement that this solution can offer, we implemented a prototype and collected a sets of data and attributes (e.g. number of packets, number of hops, transmission time, and delay). Our test bed consists of three main components. i) CSiV-hosted in Heroku cloud application platform ii) LSiV-configured in a local computer iii) SiV user-in the same network as LSiV. We evaluate the performance by using 7 HD quality videos of different lengths based on two scenarios; i) When user using SiV ii) When user is not using SiV, but regular FB/AK content delivery mechanism.

Figure 3 illustrates three conducted experiments in this regard. The first experiment (Fig. 3(a)) shows number of packets delivered in seven different HD videos that were popular during the data collection period. We fetched videos in both FB/AK and SiV solutions separately, 10 times per day (from 10:00 to 14:00) for a duration of a full week. As the figure shows, for all the video delivery, FB/AK approach uses slightly more packets compared to the proposed SiV solution. Another interesting point that we found in this experiment is that, average packet size of SiV is always larger than that of FB/AK approach.

In the second experiment we look to the total transferred traffic in the delivery procedures to understand which approach is performing better. To this end, one HD video file of 70.13 MB is used and we collect the transferred data using Wireshark.

Figure 3(b) illustrates the number of bytes (as the transferred traffic) which is transferred in the SiV and FB/AK approaches to deliver the 70.13 MB video file. The result shows FB/AK always uses more data traffic to transfer a video file, including the overhead generated due to low packet size than the SiV. On average, in-order to deliver the sample video file of the study, the FB/AK approach used 11.67% and the SiV approach used 7.01% more data to the actual size of the video. On average, 13,622 more packets were transferred in the FB/AK than in the SiV solution to deliver this video. Moreover, we found that on average, 4.4% of packets were re-transmitted in the FB/AK approach and in the SiV solution only 0.12% re-transmissions were identified.

Lastly, we analyzed total transmission time for both approaches using the previous sample HD video file. Figure 3(c) represents the time difference between first and last packet arrivals with respect to video length. Both approaches perform well in terms of the delay, but SiV transferred all the data packets in a short period of time in compare to FB/AK. Average packet transmission time for this simulation shows that SiV took only 67 sec to deliver the video whereas FB/AK used 354 sec. In summary, SiV shows better performance than the existing Facebook approach in terms of delay, total transferred traffic, and number of packets.

C. *Energy consumption optimization using SocialiVideo*

In order to reduce the power consumption across networks especially in the intermediary devices, we need to decrease the load of the traffic passing through them. When traffic increases, network devices need to route in a high data rate, increasing the power consumption of the devices. *SocialiVideo* aims to reduce number of hops between content provider and receiver and also, to reduce number of packets passing for a content delivery. As *SocialiVideo* allows nearby users to stream videos from a local storage rather than accessing remote/edge servers every time, less hops are needed when streaming from the local storage which is always less than the number of hops used in the original route. Transmission delay also proportional to the power consumption of the network elements. Higher the delay greater number of re-transmission packets and therefore, consume much more energy to deliver these excess packets. Distributing content from the local server in SiV approach reduces delay and number of re-transmitted packets as elaborated previously and hence, it reduces the energy consumption as explained below.

Based on the collected trace-route information, we calculated number of hops in both SiV and FB/AK approaches. Since each of these hops are routers/switches, we apply the model developed by Arun V. et al. [5] for the energy consumption of these devices and elaborate how much energy can be saved by using SiV solution accordingly. The total energy consumption of a router (P_{router}) depends on the idle power of the router (P_i), per packet processing

TABLE I
AVERAGE RESULTS OBTAINED FROM 10 RECORDS USING A VIDEO FILE (SIZE - 70.13 MB). (PKT STANDS FOR PACKET)

	#Pkts	Data sent (Bytes)	Pkt delivery time (Sec.)	Pkt size (Bytes)	Data rate (Bytes/Sec.)
FB/AK	86,773	78,310,012	354	904	221,342
SiV	73,151	75,084,380	67	1,031	1,119,087

energy (E_{pp}), input packet rate (N), per byte storage energy (E_{bs}), per packet switching energy (E_{ps}), and data rate (R). E_p is independent of the packet length (L), because each packet enter into the router must leave or drop. Hence, E_p is proportional to the number of packets pass through the router at that instant. However, E_s is directly proportional to the packet length. Larger the L greater the memory and also, more energy is consumed to occupy memory slots. We assume that each packet suffers equally when forwarded through a router. Based on the experimental results explained in [5], router's idle power consumption is 352W, E_{pp}=1375nJ, E_{bs}=14nJ, E_{ps}=129nJ, and the energy consumption model is as follows.

$$P_{router} = P_i + E_{pp} \times N + E_{bs} \times R + E_{ps} \times N$$

Table I summarizes the simulation results (on average) for the sample HD video file considered in the previous experiments, which compares current solution in FB/AK with SiV integrated scenario. The results show that, in-order to deliver a video content of size 70.13MB in peak hours, FB/AK and SiV approaches consume 134mJ and 126mJ respectively omitting the idle power of the router. Hence, in SiV approach, it saves an average of 8mJ in a single router. Therefore, total energy consumption when delivering the same video content from an Akamai edge network (considering 8 hops) is about 1072mJ, while SiV approach consumes approximately 126mJ. With SiV we are able to reduce 946mJ amount of energy to deliver a 70.13MB video file than the existing solution. Thus, we can conclude that, when multiple users are requesting the same content, SiV can save a considerable amount of energy compared with FB/AK content distribution mechanism. This simple experiment shows how *SocialiVideo* solution enhances the potential energy to deliver content.

V. CONCLUSION

This paper propose *SocialiVideo* as a novel solution to enhance the content delivery in CDN utilizing users social information. The main objective of this approach is to enhance the multimedia communication by providing the possibility to stream a video directly from a user's premises in the case that two parties has a social connection and both are located the same location (e.g. network, city, country). We implemented a prototype of *SocialiVideo* based on Facebook content delivery and our performance evaluation shows the proposed approach reduces access delay and network load, performs better in terms of transmission time and provides a low cost and energy efficient solution for content providers and CDNs as well as better QoE for users. *SocialiVideo* can be merged as a complementary solution to the content delivery part of a large social networks such as Facebookand be combined with the existing CDNs and data centers to enhance their data delivery.

REFERENCES

[1] M. Zhao, P. Aditya, A. Chen, Y. Lin, A. Haeberlen, P. Druschel, B. Maggs, B. Wishon, and M. Ponec, "Peer-assisted content distribution in akamai netsession," in *Internet measurement conference (IMC)*. ACM, 2013, pp. 31–42.

[2] M. P. Mills, "The cloud begins with coal: Big data, big networks, big infrastructure, and big power–an overview of the electricity used by the global digital ecosystem," *Digital Power Group*, 2013.

[3] V. Mathew, R. Sitaraman, and P. Shenoy, "Energy-aware load balancing in content delivery networks," in *INFOCOM, IEEE*, 2012, pp. 954–962.

[4] V. Sivaraman, A. Vishwanath, Z. Zhao, and C. Russell, "Profiling per-packet and per-byte energy consumption in the netfpga gigabit router," in *Computer Communications Workshops (INFOCOM WKSHPS), IEEE Conference on*. IEEE, 2011, pp. 331–336.

[5] A. Vishwanath, J. Zhu, K. Hinton, R. Ayre, and R. Tucker, "Estimating the energy consumption for packet processing, storage and switching in optical-ip routers," in *Optical Fiber Communication Conference*, 2013.

[6] S. Seyyedi and B. Akbari, "Hybrid cdn-p2p architectures for live video streaming: Comparative study of connected and unconnected meshes," in *Computer Networks and Distributed Systems (CNDS), International Symposium on*. IEEE, 2011, pp. 175–180.

[7] H. Yin, X. Liu, T. Zhan, V. Sekar, F. Qiu, C. Lin, H. Zhang, and B. Li, "Design and deployment of a hybrid cdn-p2p system for live video streaming: experiences with livesky," in *Proceedings of the 17th ACM international conference on Multimedia*. ACM, 2009, pp. 25–34.

[8] M. Afergan, T. Leighton, and J. Parikh, "Hybrid content delivery network (cdn) and peer-to-peer (p2p) network," Jul. 22 2009, eP Patent App. EP20,070,841,927.

[9] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. ACM, 2009, pp. 1–12.

[10] J. Llorca, A. Tulino, K. Guan, J. Esteban, M. Varvello, N. Choi, and D. Kilper, "Dynamic in-network caching for energy efficient content delivery," in *INFOCOM, IEEE*, 2013, pp. 245–249.

[11] A. Ruhela, S. Triukose, S. Ardon, A. Bagchi, A. Mahanti, and A. Seth, "The scope for online social network aided caching in web cdns," in *Proceedings of the ninth ACM/IEEE symposium on Architectures for networking and communications systems*, 2013, pp. 37–46.

[12] M. Cha, P. Rodriguez, J. Crowcroft, S. Moon, and X. Amatriain, "Watching television over an ip network," in *ACM SIGCOMM*, 2008.

[13] A. Brodersen, S. Scellato, and M. Wattenhofer, "Youtube around the world: Geographic popularity of videos," in *Proceedings of the 21st International Conference on World Wide Web*, 2012, pp. 241–250.

[14] M. P. Wittie, V. Pejovic, L. Deek, K. C. Almeroth, and B. Y. Zhao, "Exploiting locality of interest in online social networks," in *Proceedings of the 6th International COnference*, 2010, pp. 25:1–25:12.

[15] A. Arvidsson, M. Du, A. Aurelius, and M. Kihl, "Analysis of user demand patterns and locality for youtube traffic," in *Teletraffic Congress (ITC), 2013 25th International*, 2013, pp. 1–9.

[16] N. Laoutaris, P. Rodriguez, and L. Massoulie, "Echos: edge capacity hosting overlays of nano data centers," *ACM SIGCOMM*, 2008.

[17] W. Tang and X. Wu, "Decentralized p2p protocol for video-on-demand streaming: Simple and efficient," in *Networking, Sensing and Control (ICNSC), IEEE 12th International Conference on*, 2015.

[18] K. Suh, C. Diot, J. Kurose, L. Massoulie, C. Neumann, D. Towsley, and M. Varvello, "Push-to-peer video-on-demand system: Design and evaluation," *Selected Areas in Communications, IEEE Journal on*, pp. 1706–1716, 2007.

[19] L. Cui and N. Lu, "Socialstreaming: P2p-assisted streaming in social networks," in *Computers and Communications (ISCC), 2013 IEEE Symposium on*, 2013, pp. 000 502–000 507.

[20] R. Farahbakhsh, A. Cuevas, A. M. Ortiz, X. Han, and N. Crespi, "How far is facebook from me? facebook network infrastructure analysis," *Communications Magazine, IEEE*, pp. 134–142, 2015.

[21] M. Baldi and Y. Ofek, "End-to-end delay analysis of videoconferencing over packet-switched networks," *IEEE ACM Transactions On Networking*, vol. 8, no. 4, pp. 479–492, 2000.