OPEN ACCESS

King's Research Portal

DOI:
10.1109/ICDCS.2018.00100

Document Version
Peer reviewed version

Link to publication record in King's Research Portal

Citation for published version (APA):
Raman, A., Karamshuk, D., Sastry, N., Secker, A., & Chandaria, J. (2018). Consume Local: Towards Carbon Free Content Delivery. In *International Conference on Distributed Computing Systems* (Vol. 2018-July, pp. 994-1003). IEEE. https://doi.org/10.1109/ICDCS.2018.00100

Citing this paper
Please note that where the full-text provided on King's Research Portal is the Author Accepted Manuscript or Post-Print version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version for pagination, volume/issue, and date of publication details. And where the final published version is provided on the Research Portal, if citing you are again advised to check the publisher's website for any subsequent corrections.

General rights
Copyright and moral rights for the publications made accessible in the Research Portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognize and abide by the legal requirements associated with these rights.

•Users may download and print one copy of any publication from the Research Portal for the purpose of private study or research.
•You may not further distribute the material or use it for any profit-making activity or commercial gain
•You may freely distribute the URL identifying the publication in the Research Portal

Take down policy
If you believe that this document breaches copyright please contact librarypure@kcl.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 01. Dec. 2020

Consume Local: Towards Carbon Free Content Delivery

Aravindh Raman*, Dmytro Karamshuk‡, Nishanth Sastry*, Andrew Secker† and Jigna Chandaria†
*Department of Informatics, King's College London, UK. Email: firstname.lastname@kcl.ac.uk
‡Skyscanner, UK. Email: dima.karamshuk@skyscanner.net
†BBC R&D, London, UK. Email: firstname.lastname@rd.bbc.co.uk

***Abstract*—P2P sharing amongst consumers has been proposed as a way to decrease load on Content Delivery Networks. This paper develops an analytical model that shows an *additional* benefit of sharing content locally: Selecting close by peers to share content from leads to shorter paths compared to traditional CDNs, decreasing the overall carbon footprint of the system. Using data from a month-long trace over 3 million monthly users in London accessing TV shows online, we show that local sharing can result in a decrease of 24–48% in the system-wide carbon footprint of online video streaming, despite various obstacle factors that can restrict swarm sizes. We confirm the robustness of the savings by using realistic energy parameters drawn from two widely used settings. We also show that if the energy savings of the CDN servers are transferred as carbon credits to the end users, over 70% of users can become carbon positive, i.e., are able to support their content consumption without incurring any carbon footprint, and are able to offset their other carbon consumption. We suggest carbon credit transfers from CDNs to end users as a novel way to incentivise participation in peer-assisted content delivery.**

I. INTRODUCTION

Video streaming services are rapidly colonising the Internet and the video traffic is expected to reach 82% of all consumer Internet traffic in 2021, according to Cisco Visual Networking Index[1]. Struggling to support this growing demand for video services, content delivery networks (CDNs) are turning to clients for assistance, deploying so-called *hybrid* or *peer-assisted CDNs* [2], in which users stream video content from other peers if possible. If no suitable peers are available, users are served by the CDN's own servers, as in traditional CDNs.

The advantage of peer-assistance lies in the fact that peer-to-peer networks are innately self-sustainable because every new user contributes an upload capacity comparable to what she has consumed from the network – a self-sustainable content swarm is able to serve most of the users' requests from fellow peers, thereby offloading traffic from content servers. Indeed, Zhao et al. reports *traffic savings of 70-80%* in *Akamai NetSession* [39], and similar savings have been also suggested for *BBC iPlayer* (up to 88%) [18] and *Conviva platform* (up to 87%) [3].

In this paper, we examine a potential second advantage of using peer assistance, namely that it results in a decreased carbon footprint for content delivery. Intuitively, localising traffic to close by peers can be expected to reduce energy consumption because it requires powering fewer network hops than that of downloading a content from a distant CDN node. However, there is a fundamental trade-off: edge routers are known to be energy-inefficient in comparison to their core-counterparts [6]. Thus, although obtaining content from a peer may involve a shorter path, it may not necessarily result in energy savings as it involves traversing the edge network twice.

We study this by developing an analytical model for energy savings in peer-assisted CDNs, based on the observation that the impact of shifting costs from CDNs to peers depends on the number of participating peers. We develop a closed-form formulation linking the end-to-end energy savings achieved from peer-assistance to the average number of participating users (which we term as the *capacity* of a content swarm) as explained in Section III.

Next, we empirically analyse (Section IV) potential energy savings from peer-assistance in-the-wild using a trace of 2 billion user accesses to BBC iPlayer, one of the most widely accessed TV-on-demand platforms in the United Kingdom (UK). We focus on the subset of users in London (see Table I) and look at P2P swarms where users preferentially fetch from the closest peers that are also simultaneously consuming the same content. Using two different energy models developed by Baliga *et al.* [5] and Valancius *et al.* [34], we find that fetching content from peers yields a 24–48% reduction in the carbon footprint as compared to traditional server-based content delivery, despite taking into consideration limiting factors such as asymmetry in upload-download bandwidth, differences in bitrates required by different clients (e.g., 72 inch TV vs. mobile phone), and voluntarily limiting swarm sizes, e.g., restricting to users in the same city and the same Internet Service Provider (ISP).

Although these results clearly show the system-wide benefits of peer assisted content delivery, the arrangement is less than ideal for users who participate in the peer swarm – in effect, the users are simply being forced to take on responsibilities of content servers, and the direct beneficiary is the CDN. We therefore develop a simple carbon credit transfer scheme that transfers to each peer a volume of carbon credits equivalent to the reduction in the carbon footprint of the CDN servers due to the peer serving content to other close by users. Our analysis shows that for >70% of users, this

‡Co-first author. Work done while the author was at King's College London.
[1] http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.html

	Sep 2013	July 2014
Number of Users	3.3M	3.6M
Number of IP addresses	1.5M	1.6M
Number of Sessions	23.5M	24.2M

TABLE I: Description of the dataset.

scheme will more than offset their carbon footprint for online video streaming. Thus, carbon-free content consumption can be a strong incentive to drive the adoption of peer-assisted content delivery. The users who do not completely offset their carbon footprints with the credit transfer scheme are those who preferentially watch niche interest content, and therefore do not benefit from a large enough swarm of other peers who can upload content to them.

II. Related Papers

A comprehensive review of green networking research has been presented by Bianzino et al. in [7]. The authors suggest that the main principle which lies in the design of scalable networks, namely, *over-provisioning* and *redundancy*, goes in contradiction to the objectives of green networking. In the context of this paper, we aim to decrease usage of the core network by matching close by peers at the edge and thereby indirectly decreasing capacity provisioning requirements. The survey also creates a taxonomy for green networking enhancements, namely, *adaptive link rate*, *interface proxying*, *energy aware infrastructures* and *energy aware applications*. The analysis here shows that hybrid CDNs can be thought of as an energy aware infrastructure/application.

Two main approaches to characterise energy consumption as a function of system scale have been discussed in the literature. The *per-subscriber* approach proposed in [4], [35], [1] defines energy consumption in terms of the number of Watts consumed by individual users and characterises system scale with subscription volume. Other works, e.g., [8], [24], [34], [6], [36], [14], [15], [20], [25], take a more fine-grained *per-bit* approach and associate quantum of energy with each bit flowing through a networking node (e.g., switch, router, modem etc.), thus, measuring energy consumption as a function of the instantaneous demand. We adopted the more detailed per-bit approach in the current study because the per-session granularity of the data records we possess allows us to make fine-grain estimations of traffic demands rather than doing coarse-grain measurements based on the number of subscriptions, and also because per-user consumption patterns are highly skewed towards a small share of very active users [18]. We build on the widely used energy models of content delivery networks proposed by Valancius *et al.* [34] and Baliga *et al.* [6], adapting them for the *peer-assisted* content delivery scenario where traffic can either be delivered from active peers or from the CDN nodes if no peers are available.

The greenness of P2P overlay networks have been already discussed in the previous literature. For example, in [27] and [16] the authors provide a simplified model for comparison between energy consumption by "hot data centers" and "cool peers" and conclude that the possibility of savings depend on the baseline consumption of individual network devices and number of hops in P2P case. A similar argument is raised in [10] which extends a well known energy model by Baliga et al. [4] for the P2P case and conclude that the baseline power consumption of user's modem in idle state is a bottleneck for energy-efficient peer-to-peer networking. However, the Nano Data Centers model proposed in [34] contradicts this previous result by arguing that if a user's modem is active for a peer who is currently online there would be no baseline cost for also sharing content with other peers in the network. This is particularly true when users only share the content which they are currently watching. The carbon footprint of the hybrid peer-assisted deployment is considered in [25] where the authors indicate a positive scaling effect of P2P sharing on energy savings. With respect to these previous works, our contribution lies in estimating the end-to-end energy savings for various demand patterns in peer-assisted CDNs and devising a carbon-credit transfer schema for incentivising users to participate. Additionally, we conduct a large scale empirical analysis of energy savings for BBC iPlayer.

In a broader context, peer-assisted content delivery has been thoroughly surveyed by Lu *et al.* [23] and Anjum *et al.* [2]. The main focus of this flow of papers lies in understanding traffic savings (rather than energy savings) via large-scale analysis of users' sessions in Akamai NetSession [13][39], Conviva [3], BBC iPlayer [18], Kankan [38], Todou [22] and Spotify [11]. With respect to these previous works on understanding performance of peer-assistance, in this paper we focus on sustainability aspects of hybrid content delivery rather than traffic savings and consider both end-to-end greenness of peer-assistance as well as potential impact on individual users.

III. Energy Savings Model

We aim to understand potential energy savings from peer-assisted content delivery. We tackle this problem by measuring how much energy, which otherwise would be consumed by delivering content from the CDN, can be saved if the content is instead delivered from nearby peers. The intuition, which we explore in detail in the rest of this section, is that as a result of fewer hops and fewer equipments, the energy of delivering traffic from nearby online peers is generally lower than that of powering and delivering content from distant CDN nodes. Furthermore, as the size of the peer-to-peer swarm grows, the density of users in the network increases, which allows users to be matched with other peers who are closer by, leading in turn to an increase in the energy savings.

A. General strategy and terminology

Our goal is to compute the potential savings resulting from taking a hybrid peer-assisted approach as compared to a traditional server-based CDN. Our general strategy will be to calculate the energy consumption as per-bit energy cost functions for delivering content to users from CDN servers ($\Psi_s(\cdot)$), and from other peers ($\Psi_p(\cdot)$).

Given a user who consumes T_u bytes of content, the energy required in traditional server-based CDNs is simply $\Psi_s(T_u)$. When a peer-assisted hybrid CDN strategy is deployed, content can be delivered to a user either from a content delivery node (i.e., from a server) or from other users in the network (i.e., peers). We compute the total energy by computing the fraction G of the total traffic T_u which can be offloaded to peers. Thus the energy required for the hybrid CDN case will be $\Psi_p(GT_u) + \Psi_s((1-G)T_u)$.

With this, we can compute energy savings from taking a hybrid peer-assisted CDN approach as:

$$S = 1 - \frac{\Psi_s((1-G)T_u) + \Psi_p(GT_u)}{\Psi_s(T_u)} \quad (1)$$

Note that the savings can be negative if the hybrid approach consumes more energy. The rest of this section is devoted to obtaining a closed form formula for S.

B. Measuring the scale of peer assistance

We wish to study how energy savings evolve as the system scales. We use the average number of peers in the system to measure the scale of the system. We term this as the *swarm capacity* or *peer capacity*. With more users in the swarm, there are more peers to upload content to other peers, hence we also interchangeably use the term *peer upload capacity* or simply *capacity*.

The fact that the capacity of a P2P swarm increases in proportion with the number of users in the swarm has been termed as "self-scaling" (e.g., [9]). Following Menasche *et al.* [26], we model this self-scaling property of peer-to-peer swarms by treating each swarm as a $M/M/\infty$ queuing system with infinite servers: users who arrive at a swarm do not wait to be serviced, and can be served instantly by other members of the swarm. A user who arrives when the swarm is empty (or when there are too few peers to sustain swarming) departs immediately without being serviced by the swarm (In our case, this user is instead served by the edge servers of the CDN, and starts a new swarm).

Consider a swarm for sharing a content item. Since there is no queuing time, the average time spent by users in the system is simply the average time spent watching the content, u. If users arrive at an average rate r, then according to Little's law, the average number of users in the swarm can be written as

$$c = ur$$

. We term c as the *capacity* of the swarm.

C. How much traffic can be offloaded to peers

We develop a simple analytical model to understand what fraction G of traffic can be offloaded to peers if traditional CDN servers are enhanced with peer-assistance.

We divide the swarm into small time windows of size $\Delta\tau$. We assume that users are streaming content from the system rather than aggressively buffering it and, therefore, at each $\Delta\tau$ each user downloads a content buffer of length $\Delta\tau$ (or equivalently $\beta\Delta\tau$ bytes if β is the bitrate of the content).

Variable	Description
S	energy savings from peer-assisted content delivery
G	traffic savings, i.e., share of traffic offloaded to peers
T_u	useful traffic, i.e., total amount of bytes watched by users
c	capacity (i.e., average number of users) of a content swarm
r	peer arrival rate in a content swarm
u	average session duration in a content swarm
p	probability of having at least one user online
L	instantaneous number of peers in a swarm
β	bitrate of a content
q	upload bandwidth of users
ψ_s	per-bit energy consumption of a traditional CDN
ψ_p	per-bit energy consumption of a peer-assisted CDN
$\Delta\tau$	size of a time window
ΔT_u	traffic watched by all users in a swarm during $\Delta\tau$
ΔT_p	traffic downloaded from peers in a swarm during $\Delta\tau$

TABLE II: Parameters of the analytical model (Also see Tables III and IV for other symbols)

Let the total traffic requirement of the swarm during $\Delta\tau$ be ΔT_u bytes, of which $\Delta T_p < \Delta T_u$ bytes is traffic offloaded to peers, and the traffic load to the server is $\Delta T_u - \Delta T_p$. Suppose there are L active users during the time window. The content can be broken down into chunks, and each chunk can be downloaded by *one* of the L downloaders interested in the content item, who can then share with the other $L-1$ users [2].

In this scheme, the collective traffic requirement of the L downloading users is

$$\Delta T_u = L\beta\Delta\tau$$

If each user has an upload bandwidth q, the amount of traffic shared during the time window between peers is

$$\Delta T_p = \begin{cases} (L-1)q\Delta\tau & \text{if } L > 1 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Summing across all time windows of a total duration $\sum \Delta\tau$, we get

$$\sum \Delta T_u = c\beta \sum \Delta\tau$$

and

$$\sum \Delta T_p = (c-p)q \sum \Delta\tau$$

where c is the average number of users in the system (also known as the capacity of the swarm) and p denotes the probability of having at least one user online which for $M/M/\infty$ queue is known to be $p = 1 - e^{-c}$. This allows us to estimate traffic offloaded to peers with the following equation[3]:

[2]We assume *managed swarming* similar to AntFarm [29] or Akamai NetSession [39], where a central server efficiently manages which peer is matched with which other peer, and also which peer gets which bytes from the server and which bytes from other peers. Thus, problems such as rare chunks, possible in BitTorrent-like swarms [19], are not a concern to us.

[3]Note that for the content swarms with the expected number of users online $c = 1$, there is still a non-trivial probability for content sharing (i.e., having more than one user online) within $\Delta\tau$ as the users join the system in Poisson fashion. Therefore, opportunities are for offloading $G = 0.37\frac{q}{\beta}$.

Layer	Count	Localisation Probability
Exchange Point	345	p_{exp} = 0.29 %
Point of Presence	9	p_{pop} = 11.11 %
Core Router	1	p_{core} = 100 %

TABLE III: Probability of localising peers within a given layer of ISP metropolitan network laid out as in Fig. 1. The counts represent numbers of exchange points, points of presence and nationwide core routers for a major ISP in London, obtained through private conversations with the ISP. The localisation probability is the probability that a given peer will be under a given node at the given layer.

$$G = \frac{\sum \Delta T_p}{\sum \Delta T_u} = \frac{q}{\beta} \frac{c + e^{-c} - 1}{c} \tag{3}$$

D. Per-bit cost function for P2P and CDN traffic

Given the proportion G of traffic that can be offloaded to peers in a particular swarm, we wish to translate this into energy consumption. To accomplish this, we turn to energy models in the literature, which provide *per-bit* and *per-hop* energy consumption values, based on actual measurements [34], or using data-sheets from real equipment [4]. These models allow us to calculate energy consumption proportional[4] to the number of bytes T transferred through the network with a proportionality factor ψ, i.e., $\Psi(T) = T\psi$, with $\psi = \psi_s$ for serving users from CDN servers, and $\psi = \psi_p$ for peer-peer traffic. Note that because we are interested solely in the *difference between* energy consumed between server-based content delivery and a hybrid peer-to-peer assisted case, the models below do not explicitly consider end-user equipment (e.g., laptop vs. 5 inch mobile phone vs. 72 inch TV), since the same device is used regardless of whether the content is obtained from a server or another peer.

1) Per-bit energy cost for delivering from servers (ψ_s): Following the Valancius [34] and Baliga [4] models, we can straightforwardly define the per-bit energy consumption model for delivering data from CDN servers as:

$$\psi_s = PUE\,(\gamma_s + \gamma_{cdn}) + l\gamma_m \tag{4}$$

where γ_s, γ_m and γ_{cdn} are the the per-bit energy consumption of the CDN node (γ_s), end-user's modem or other customer premises equipment not shared with other users (γ_m), and networking equipment between a user and a CDN node (γ_{cdn}) respectively. PUE is the power usage efficiency metric of the network which accounts for redundancy and l is the energy "loss" for end-user equipment.

2) Per-bit cost function for P2P delivery (ψ_p): Similarly, the per-bit energy consumption model for delivering content from peers can be defined as:

$$\psi_p = \psi_p^m + \psi_p^r \tag{5}$$
$$= 2l\gamma_m + PUE\,\gamma_{p2p} \tag{6}$$

where $\psi_p^m = 2l\gamma_m$ is the per-bit consumption of the user premises equipment and is independent of the size of the swarm. ψ_p^r = PUE γ_{p2p} is the swarm-size dependent part that depends on the length of the routes between peers who are assisting each other. ψ_p^m is counted twice to account for simultaneous downloading and sharing (uploading) the content with other peers. The impact of the route length within the network (i.e., after end user equipment) on the per-bit energy consumption for carrying P2P traffic is reflected in γ_{p2p}. Note that unlike γ_{cdn}, γ_{p2p} varies depending on the size L of the content swarm. Intuitively, the bigger the swarm, the higher the chance to find peers close by in the network, so the smaller γ_{p2p} is.

We can estimate the energy consumed on network equipment $\gamma_{p2p}(L)$ in a time window with L online users as follows: As discussed previously, we assume that peers are matched with others inside the same ISP and that from the perspective of the user, an ISP has a tree-like topology schematically represented in Fig. 1[5]. From our conversations with a large national-scale ISP operating in the considered region, there are $n_{exp} = 345$ exchange points, $n_{pop} = 9$ points of presence (PoP), and 1 core router (See Table III). Thus if we pick one of the users consuming a content item, the probability that it would be under a particular exchange point (resp. PoP or core router) would be $p_{exp} = 1/n_{exp}$ (resp. $p_{pop} = 1/n_{\text{pop}}$, $p_{core} = 1/n_{core}$).

Consider a user in a swarm of L users who is under a particular exchange point (resp. PoP or core router). The probability of finding a local peer under that exchange point (resp. PoP or core router) can be written as a function of p_{exp} (resp. p_{pop} or p_{core}) as $P_{exp}(L) = 1 - (1 - p_{exp})^{L-1}$. Since finding a peer lower down in the hierarchy (i.e., closer by in network distance) is preferred, we can write

$$\begin{aligned}\gamma_{p2p}(L) = &\gamma_{exp}(P_{exp}(L)) \\ &+ \gamma_{pop}(P_{pop}(L) - P_{exp}(L)) \\ &+ \gamma_{core}(P_{core}(L) - P_{pop}(L))\end{aligned} \tag{7}$$

where γ_{exp}, γ_{pop} and γ_{core} are the per-bit energy consumption numbers for paths localised to an exchange point, PoP and the core respectively.

This is an approximation which is based on the *expected* distance between pairs of users who may be matched by a centralised swarm manager, given a swarm of a certain size. In general, $\gamma_{p2p}(L)$ varies based on the algorithm used for matching peers. However our empirical analyses (Section IV) suggest that this approach gives a good approximation of $\gamma_{p2p}(L)$.

[4]This assumes energy proportional equipment and may therefore not be valid for low traffic volumes T. However, such models can be reasonably accurate, as shown through measurements in [34].

[5]This schema, as well as the numbers of equipments in each stage in the hierarchy, are based on private conversations with a large national-scale ISP in the considered city.

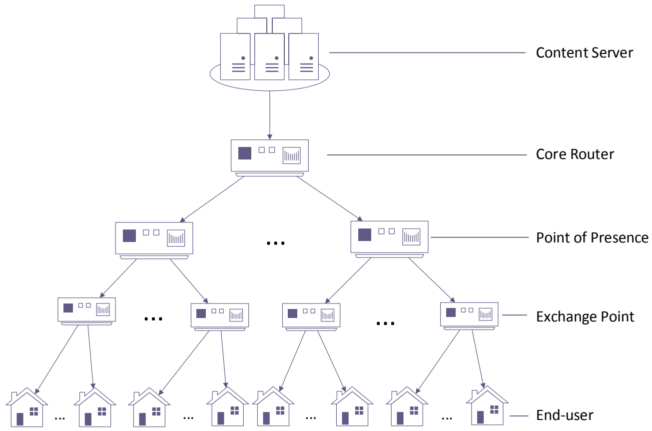

Fig. 1: Metropolitan network topology (verified through private conversations with a large national ISP which carries the traffic of this TV streaming application.).

Variable	Valancius, nJ/bit	Baliga, nJ/bit
Content Server (γ_s)	211.1	281.3
End User Modem (γ_m)	100.0	100.0
Traditional CDN Network (γ_{cdn})	1050.0	142.5
P2P Network within ExP (γ_{exp})	300.00	144.86
P2P Network within POP (γ_{pop})	600.00	197.48
P2P Network within Core (γ_{core})	900.00	245.74
Power Efficiency (PUE)	1.2	1.2
End-user energy loss (l)	1.07	1.07

TABLE IV: Energy parameters as measured by Valancius *et al.* [34] and Baliga *et al.* [6]. For Valancius *et al.* the network parameters are calculated as $h \times 150$ nJ/bit where h is the number of network hops between sender and receiver, i.e., $\gamma_{cdn} = 7 \times 150$ nJ/bit, $\gamma_{core} = 6 \times 150$ nJ/bit, $\gamma_{pop} = 4 \times 150$ nJ/bit, $\gamma_{exp} = 2 \times 150$ nJ/bit. For Baliga *et al.* the network parameters are calculated as a sum of consumption of all individual networking nodes (e.g., routers, switches) between end-user and server nodes (CDN) and between peers localized either within an Ethernet switch (ExP), within an edge router (PoP) or within a core router (Core). Values for power efficiency and end-user energy loss are taken from Valancius *et al.*for consistency.

E. Total energy savings

We are now able to compute the energy savings obtained by using the peer-assisted approach. When a client is served from a CDN server the average distance between the user and server remains the same regardless of the content item or the number of users. Thus the energy consumed for content item of size T bytes is simply $\Psi_s(T) = \psi_s T$. In contrast, in the P2P case, the per-bit energy consumption in the network, γ_{p2p}, depends on the distance between peers in the network, which in turn depends on the size of the content swarm. Thus, to measure P2P consumption we split the energy cost function into a swarm-size dependent component $\Psi_p^r(T)$ for networking equipment and swarm-size oblivious $\Psi_p^m(T) = T\psi_p^m$ for the user's modem energy consumption, i.e., $\Psi_p(T) = T\psi_p^m + \Psi_p^r(T)$. Putting these results in Eq. 1 for energy savings and using Eq. 3 for traffic gain G, we obtain:

$$\begin{aligned} S &= G\frac{(\psi_s - \psi_p^m)}{\psi_s} - \frac{\Psi_p^r(T)}{\psi_s T_u} \\ &= \frac{q(c + e^{-c} - 1)(\psi_s - \psi_p^m)}{\beta c \psi_s} - \frac{\Psi_p^r(T)}{\beta c \psi_s \sum \Delta\tau} \end{aligned} \quad (8)$$

To calculate the swarm size dependent $\Psi_p^r(T)$ we use the per-bit energy cost function ψ_p^r from Eq. 6, sum across all the bytes transferred between peers within a time window as computed from Eq. 2, and then further aggregate across all time windows. i.e.: $\Psi_p^r(T) = \sum \psi_p^r \Delta T_p$. Expanding, we get:

$$\Psi_p^r(T) = \sum \gamma_{p2p}(L) \times \text{PUE} \times (L-1) \times q\Delta\tau \quad (9)$$

$$\begin{aligned} &= q \times PUE \times \sum \Delta\tau \\ &\quad [(\gamma_{pop} - \gamma_{exp}) f(p_{exp}, c) \\ &\quad + (\gamma_{core} - \gamma_{pop}) f(p_{pop}, c) \\ &\quad + \gamma_{core} \times f(p_{core}, c)] \end{aligned} \quad (10)$$

where

$$f(p, c) = \begin{cases} e^{-c} + c - 1 & \text{if } p = 1 \\ \frac{e^{-cp}(1-c+cp) - e^{-c}p}{1-p} + c - 1 & \text{otherwise} \end{cases} \quad (11)$$

Note that here we again used the result for the $M/M/\infty$ queuing system to estimate the sum $\sum \gamma_{p2p}(L) \sim E[\gamma_{p2p}(L)]$ where $E[\gamma_{p2p}(L)]$ is the expected value of γ_{p2p}. By definition $E[\gamma_{p2p}(L)] = \sum \gamma_{p2p}(L) f_L(L)$ where $f_L(L)$ is the probability distribution function of having exactly L users in a time window and it is known to be a Poisson distribution with the expected value c (i.e., capacity of the swarm) for the $M/M/\infty$ queue. This brings us to the final master equation derived by substituting $\Psi_p^r(T)$ in Eq. 8:

$$\begin{aligned} S = \frac{q(c + e^{-c} - 1)(\psi_s - \psi_p^m)}{\beta c \psi_s} - \frac{q \times PUE}{\beta c \psi_s} \times \\ [(\gamma_{pop} - \gamma_{exp}) f(p_{exp}, c) \\ + (\gamma_{core} - \gamma_{pop}) f(p_{pop}, c) \\ + \gamma_{core} f(p_{core}, c)] \end{aligned} \quad (12)$$

IV. Empirical Analysis

To explore the savings realisable and to understand how well Eq. 12 can approximate reality, we examine a large real-world workload that includes accesses to a leading on-demand streaming platform in the United Kingdom, BBC iPlayer, for an year from July 2013 to July 2014. BBC iPlayer is accessible only from the country's IP addresses[6], and the trace covers the equivalent of 40% the UK's population with a mean of 32

[6]Although there are known ways to break such restrictions, e.g., using VPN end points, we believe that these constitute a minority of accesses in comparison with the volume of accesses within the country.

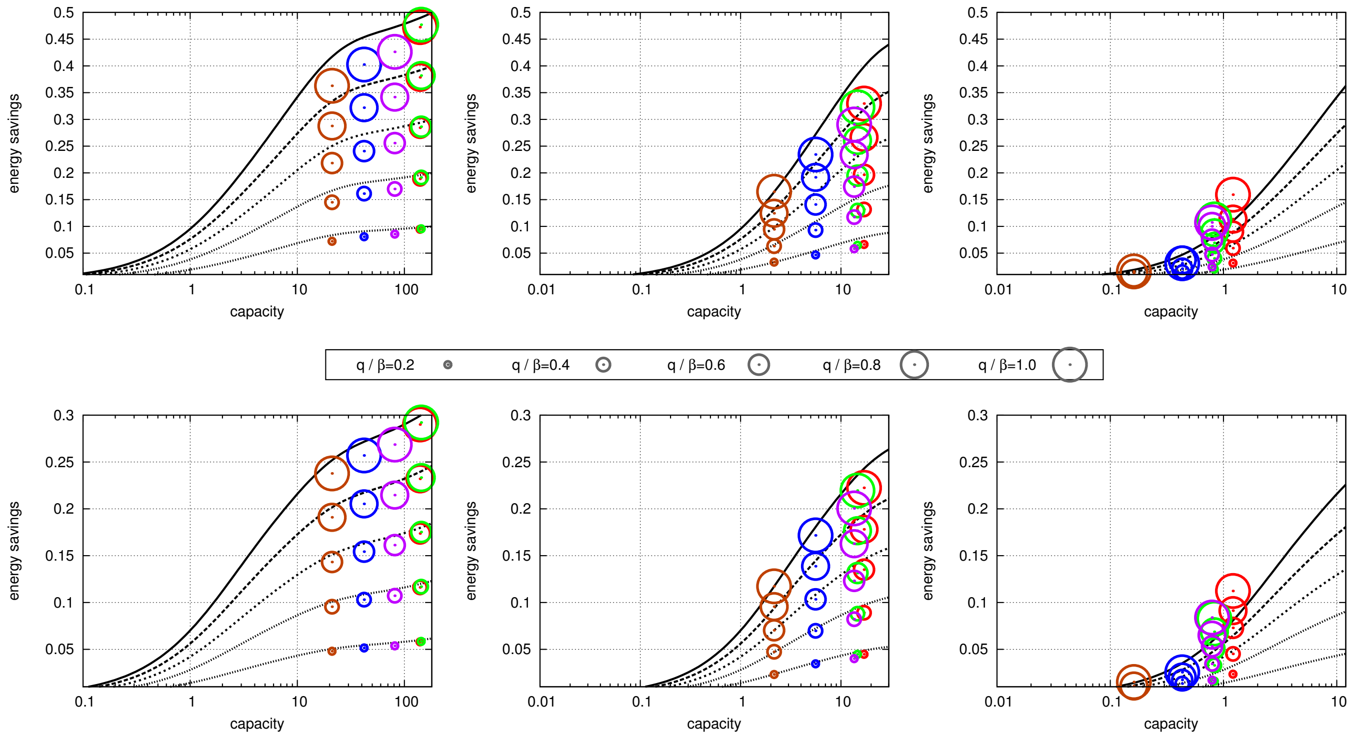

Fig. 2: Energy savings estimated theoretically (black curve) and via simulations (dots), for exemplar highly popular (Left col.), medium popular (Centre col.) and unpopular (Right col.) content items, across top 5 ISPs (different colours) for energy parameters from Baliga *et al.* (bottom row) and Valancius *et al.* (top row).

million accesses per month. Focusing on users from London, we tease apart the energy savings realisable using the hybrid CDN approach.

A. Simulation and dataset description

BBC iPlayer[7] is a widely used video streaming application and is a catch-up TV service that makes available for on-demand streaming most of the programmes broadcast on TV channels across the UK. The application is available for both web and mobile platforms and competes with the likes of YouTube and Netflix in traffic volume[8].

Unlike YouTube (but like Netflix), BBC iPlayer hosts ad-free content, and TV shows are much longer than the average YouTube video. Our data reported the equivalent of over 40% of the country's population accessing the application in a representative month. We focus on London, a large city in the UK, where the number of users in different months covers 36–41% the population of the city (Table I).

Although BBC iPlayer is currently an over-the-top streaming service using traditional CDNs, we use trace-driven simulations to explore the potential advantage of a hybrid P2P CDN in comparison with a streaming-only CDN. To aid our analysis, we implemented a discrete time step simulator where timestamps of events (i.e., start times and durations), and bitrates of user sessions, are taken from the trace. The simulator proceeds with a fixed time step of $\Delta\tau = 10$ seconds where for each $\Delta\tau$ the simulator assesses how many peers are online, how much upload bandwidth they can share and how much download bandwidth they require to stream the content. The calculations are then done for the number of bytes that would be streamed from content servers and from peers, correspondingly. We match peers that are closest to each other, and calculate energy savings S_{sim} obtained from simulations. We then compare S_{sim} with the theoretical S_{theo} (Eq. 12).

B. Energy savings in-the-wild

1) Factors to consider: Eq. 12 clearly suggests that the energy savings from hybrid CDNs depends on three classes of factors: (*i*) parameters of the energy model, (*ii*) the size of the swarm, i.e., the number of peers who can share content (*iii*) the capacity of the peers sharing to upload content to their peers. We examine the impact of each of these factors in Fig. 2. Below we discuss (in reverse of the above order) how we study these factors:

[7]https://www.bbc.co.uk/iplayer

[8]http://mediatel.co.uk/newsline/2014/03/28/nielsen-data-report-february-2014

Upload bandwidth is not a limitation It is commonly perceived that P2P is limited by the asymmetry in upload/download bandwidth, due to the use of technologies such as Asymmetric DSL (aDSL) to reach consumer premises. However, this is largely a myth in today's networks – due to continuous improvements in broadband speeds, upload bandwidths in today's homes are more than sufficient to support P2P swarms for bitrates commonly used by streaming platforms. For instance, Netflix recommends a *download* broadband speed of 1.5Mbps[9], whereas the average *upload* speeds in the UK are around 4.3Mbps[10]. *The most common bitrate in BBC iPlayer is 1.5Mbps [28], which can easily be supported using P2P in today's broadband networks*. In other words, a user wanting to access Netflix content can be served by one of its peers which uploads the content to the first user. Indeed, current speeds are almost sufficient to even sustain upload of an SD video stream. At the current rate of improvements in upload speeds, sustained SD video uploads by peers for real-time streaming should be possible by the next year or two. In today's networks, SD streams can be sustained if two or more peers collaborate to upload to a single other peer. We incorporate limitations in upload bandwidth q relative to the bitrate β of the streaming application by considering their ratio q/β as a variable parameter in Fig. 2.

Factors affecting swarm size The most important factor affecting the size of the swarm is the popularity of the content item. We consider three different content items with various levels of popularity and hence swarm sizes: an episode of a highly popular "Bad Education" series which accounts for over 100K views in September 2013 (Left column in Fig. 2), an episode of a series with an intermediate popularity level of around 10K views, "Question Time" (Centre column), and an unpopular item "What's to Eat" with around 1K views (Right column).

Within each content item, the swarm of consumers for each content item is further split based on average bitrates (a user watching on a modern internet-connected HD TV with a large screen may find it difficult to stream from a peer who is watching at a lower bitrate on her mobile phone). Further, participants in a swarm are limited to those who are streaming the content item at the same time. Finally, we consider ISP-friendly P2P swarming and always match users with other peers within the same ISP. Whilst this last factor is not always a necessary constraint, it has been suggested in various recent twists on peer-assisted streaming (e.g., see references in [30]), and can help avoid potential throttling by ISPs. Because it limits the size of the content swarms drastically, ISP-friendly P2P swarms can provide a *lower* bound on achievable savings. The size of the swarm after all these factors are considered is taken as c and the corresponding energy savings are calculated.

Parameters of the energy model The savings achieved clearly depend on how we calculate and account for the energy consumed by each flow. Eq. 12 computes the energy savings in terms of "per-bit" energy consumed. Quantifying the exact amount of energy consumed by a bit or by a particular flow in today's Internet is a difficult problem, although several attempts have been made to characterise this figure. To understand how the precise energy model used impacts potential savings, and mitigate the uncertainty in the exact savings calculated, we use two independently developed, widely used and widely cited models, by Valancius *et al.* [34] and by Baliga *et al.* [4]. The parameters for these models are listed in Table IV. Both sets of parameters are based on direct measurements of energy consumption in real networking equipment and/or data sheet information from commonly used routers, and closely fit the intended use case of video-based content delivery. Although the "true" energy consumed by a flow may vary from the ones reported in this paper because of approximations made or a different set of networking equipment used, we believe that the savings calculated by the parameters of the two models and the variation between the headline figures calculated using these parameters provides a good indication of the "real" energy savings in any given instantiation of a hybrid peer-assisted CDN. More importantly, it allows us to understand how the energy savings vary with factors such as swarm size. Furthermore, we only calculate the *percentage* savings in energy; and to calculate this, we only require the calculated energy to be roughly proportional to the "actual" energy consumed.

[9] https://help.netflix.com/en/node/306

[10] http://www.ispreview.co.uk/index.php/2017/04/ofcom-2017-study-average-uk-home-broadband-speeds-rise-36-2mbps.html

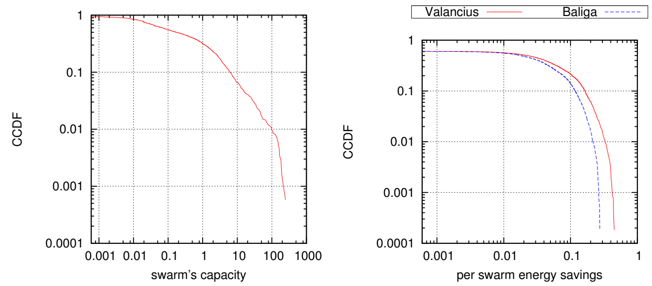

Fig. 3: Distribution of per-swarm capacities (left) and energy savings (right) across all content items in the content catalogue.

2) Understanding the magnitude of energy savings: Fig. 2 shows that for popular items (left column), there are considerable savings across all ISPs (35–48% according to Valancius *et al.*; 24–29% according to Baliga *et al.*), and savings remain at over 10% in both models even when the upload bandwidth is at an unrealistically low value of 0.4 of the content bitrate (i.e., $q/\beta = 0.4$). By contrast, the savings for the unpopular item (right column) are always less than 10%, and the intermediate popularity item (middle column) generally sees savings of 5 – 20% depending on the ISP and the ratio q/β. Notice also that the black lines estimated using Eq. 12 are generally in good agreement with the simulation, indicating that our formula is a reasonable approximation that can potentially be used for network planning purposes.

The next logical step is to ask how the energy savings are distributed across the content corpus. As shown in Fig. 3 (left), the catalogue of items available for on-demand streaming consists of a few popular items but a large majority of unpopular items. This results in highly disproportionate savings for the popular items as compared with the majority of content items (Fig. 3 (right)) – median per-item savings are around 2% for both the Valancius *et al.* and Baliga *et al.* models, whereas the Top-1% of the popular items obtain over 21% (resp. 33%) of energy savings using energy parameters from the Baliga *et al.* (resp. Valancius *et al.*) model.

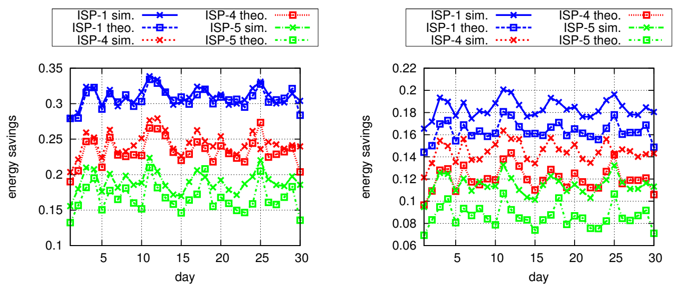

Fig. 4: The aggregate energy savings with parameters from Valancius et al. (left) and Baliga et al. (right) across various ISPs throughout the month of Sep 2013 achieved with data-drive simulations (sim.) and from analytical analysis (theo.).

We then ask how the energy savings add up for the whole system, since the popular items, which yield the best savings, also obtain the most accesses. Fig. 4 presents the result of this aggregate analysis, showing the daily savings across all requests to all items in the content catalogue, measured across a whole month. Despite some daily fluctuations, on average around 30% (18%) of energy savings can be achieved for the biggest ISP with the Valancius *et al.* (Baliga *et al.*) model, suggesting that the popular items are able to compensate for the small savings from the majority of unpopular items. Again, the simulation results match the theory from Eq. 12.

V. Carbon credit transfers

The previous analysis indicates that the system becomes greener as a whole by using peer-assisted content delivery. However, this involves *end users* taking on content delivery tasks, and increasing their energy consumption (and energy bills), and therefore they need to be compensated or incentivised for assisting CDN servers. While there are many possible incentive schemes, in this section, we consider a carbon credit transfer: *we ask whether the carbon footprints of users' content consumption patterns can be decreased or eliminated as a result of using peer-assisted CDNs, by passing carbon credits from the CDN to end users.*

The central idea is illustrated in Fig. 5. As swarm capacity increases, P2P content distribution takes over and the energy savings of CDN servers increase. This leads to end-to-end system savings of the system as a whole, but simultaneously, the collective energy consumed by the users increases as well

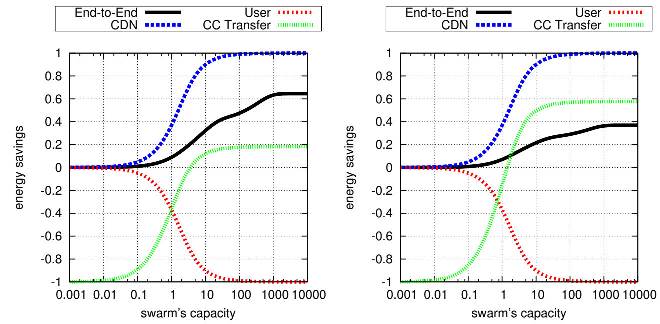

Fig. 5: Energy savings in the network with different parameters (Valancius et al. (left) and Baliga et al. (right)) as a function of swarm's capacity. Energy savings for CDN and Users are normalized by the corresponding energy costs of CDN and Users when peer-assistance is disabled. The end-to-end curve shows the energy savings of the system as a whole. CC transfer indicates the collective carbon footprint of users after the energy savings of the CDN has been transferred to the users.

(i.e., user savings decrease). *Carbon credit transfer* involves counting the savings accrued by CDN servers as a carbon credit, and using them to alleviate the increased carbon footprint of the end users watching the video.

Given the share G of traffic offloaded to peers and per-bit server consumption γ_s we measure the per-bit energy saved by CDNs as $G\,\gamma_s$. Similarly, users consume an overall energy $l(1+G)\,\gamma_m$ for downloading and sharing content. Therefore, we estimate the normalised carbon credit transfer from CDNs to users as:

$$CCT = \frac{PUE\,\gamma_s\,G - l\,\gamma_m\,(1+G)}{l\,\gamma_m\,(1+G)} \tag{13}$$

Naturally, when a user does not share, or equivalently, if there are no other users online, $CCT = -1$ (CCT is shown as a green line in Fig. 5). As swarm size increases, the energy savings of the CDN increase, and so it can pass on more carbon credit to participating end users. An end-user becomes *carbon neutral* when $CCT = 0$ or equivalently when

$$G = (PUE\,\gamma_m)/(PUE\,\gamma_s - l\,\gamma_m).$$

Beyond this point, users become "carbon positive" and can effectively use the transferred carbon credits to offset their other carbon emissions. In the asymptotic case when $G = 1$, end users are carbon positive by 18% (58%) of their total content consumption energy footprint in the Valancius *et al.* (Baliga *et al.*) model.

Fig. 6 plots the distribution across all users in the trace of their *net* carbon footprint after carbon credit transfer as defined in Eq. 13. Significant amount of users (around 41% users in Valancius *et al.* and more than 70% for Baliga *et al.*) benefit, and become carbon positive. The users who remain carbon negative are those who mostly watch niche interest content items whose swarm sizes are too small.

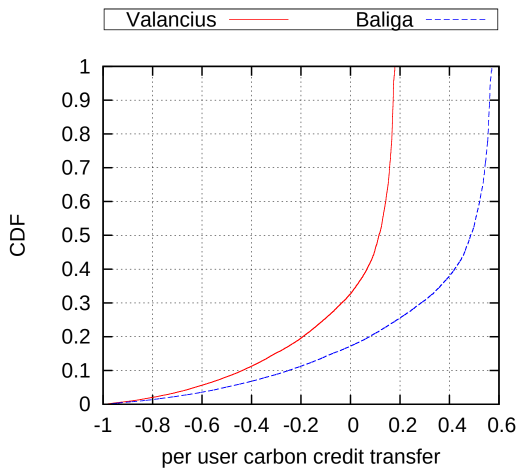

Fig. 6: Distribution of per-user carbon footprints according to the Valancius *et al.* and Baliga *et al.* models for all users across London, after carbon credit transfer from the CDN.

VI. Conclusion

Many Content Delivery Networks (CDNs) have adopted features drawn from peer-to-peer (P2P) networks, allowing users to download content from each other rather than from CDN servers. While this provides distinct advantages for the CDN provider, by decreasing its traffic costs, and costs of provisioning for peak loads, users and Internet Service Providers (ISPs) have little direct incentive to participate. Indeed, previous studies from a major CDN provider Akamai have shown that as little as 30% of its users participate by contributing upload capacity [39]. Similarly, ISPs may object if the CDN provider matches peers from different ISPs, as exchanging content between them can cause an increase in the ISPs' transit traffic costs. Besides this, other issues such as the need to match peers downloading content at the same bitrate, and typical asymmetries in upload-download bandwidths, can limit the gains that can be seen from hybrid CDNs. Extending previous studies (e.g., [18]) which showed that *traffic* gains can be had despite such obstacle factors, this paper showed that there are system-wide reductions of 24–48% in the *carbon footprint* of online video streaming, despite making the P2P swarms ISP friendly, and restricting swarms to users within the same ISP. We also considered incentivising the users, and showed that by transferring the savings achieved by the CDN to the users, users' carbon footprint from online video streaming can be completely offset as compared to not sharing their content in a P2P swarm. We offer carbon neutral (or indeed carbon positive) online video streaming as a good incentive for users to participate in hybrid CDN. Future work can extend the proposed model by incorporating predititive preloading techniques [17], [33], by adding caching schemes [31], [12], by considering live video streaming scenarios [32] and by building a viable economic model of user behaviour [37], [21].

Acknowledgment

The work is partially supported by the EU-India REACH Project under Grant No. ICI+/2014/342-986, by UK Engineering and Physical Sciences Research Council (EPSRC) via the Internet of Silicon Retinas Project, Grant No. EP/P022723/1 and by the 5GUK testbeds & trials from DCMS.

References

[1] S. Aleksić and A. Lovrić. Power Consumption of Wired Access Network Technologies. In *7th International Symposium on Communication Systems Networks and Digital Signal Processing*, pages 147–151. IEEE, 2010.

[2] N. Anjum, D. Karamshuk, M. Shikh-Bahaei, and N. Sastry. Survey on Peer-assisted Content Delivery Networks. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 116(C):79–95, 2017.

[3] A. Balachandran, V. Sekar, A. Akella, and S. Seshan. Analyzing the Potential Benefits of CDN Augmentation Strategies for Internet Video Workloads. In *Proceedings of the IMC 2013*, pages 43–56. ACM, 2013.

[4] J. Baliga, R. Ayre, K. Hinton, W. V. Sorin, and R. S. Tucker. Energy Consumption in Optical IP Networks. *Journal of Lightwave Technology*, 27(13):2391–2403, 2009.

[5] J. Baliga, R. Ayre, K. Hinton, and R. Tucker. Energy Consumption in Wired and Wireless Access Networks. *IEEE Communications Magazine*, 49(6):70–77, 2011.

[6] J. Baliga, R. W. Ayre, K. Hinton, and R. Tucker. Green Cloud Computing: Balancing Energy in Processing, Storage, and Transport. *Proceedings of the IEEE*, 99(1):149–167, 2011.

[7] A. P. Bianzino, C. Chaudet, D. Rossi, and J. Rougier. A Survey of Green Networking Research. *IEEE Communications Surveys & Tutorials*, 14(1):3–20, 2012.

[8] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright. Power Awareness in Network Design and Routing. In *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*. IEEE, 2008.

[9] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker. Making gnutella-like P2P systems scalable. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 407–418. ACM, 2003.

[10] A. Feldmann, A. Gladisch, M. Kind, C. Lange, G. Smaragdakis, and F. Westphal. Energy Trade-offs among Content Delivery Architectures. In *Telecommunications Internet and Media Techno Economics (CTTE), 2010 9th Conference on*, pages 1–6. IEEE, 2010.

[11] M. Goldmann and G. Kreitz. Measurements on the Spotify Peer-Assisted Music-on-Demand Streaming System. In *IEEE International Conference on Peer-to-Peer Computing (P2P)*, pages 206–211. IEEE, 2011.

[12] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire. Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution. *IEEE Communications Magazine*, 51(4):142–149, April 2013.

[13] C. Huang, A. Wang, J. Li, and K. W. Ross. Understanding Hybrid CDN-P2P: Why Limelight Needs its Own Red Swoosh. In *Proceedings of the 18th NOSSDAV workshop*, pages 75–80. ACM, 2008.

[14] F. Jalali, R. Ayre, A. Vishwanath, K. Hinton, T. Alpcan, and R. Tucker. Energy Consumption Comparison of Nano and Centralized Data Centers. *Proc. of Greenmetrics. Austin, Texas, USA: ACM*, 2014.

[15] F. Jalali, K. Hinton, R. Ayre, T. Alpcan, and R. S. Tucker. Fog Computing May Help to Save Energy in Cloud Computing. *IEEE Journal on Selected Areas in Communications*, 34(5):1728–1739, 2016.

[16] C. Jarabek and M. Wang. Can P2P Help the Cloud Go Green? In *Performance Computing and Communications Conference (IPCCC), 2011 IEEE 30th International*, pages 1–10. IEEE, 2011.

[17] D. Karamshuk, N. Sastry, M. Al-Bassam, A. Secker, and J. Chandaria. Take-Away TV: Recharging Work Commutes With Predictive Preloading of Catch-Up TV Content. *IEEE Journal on Selected Areas in Communications*, 34(8):2091–2101, Aug 2016.

[18] D. Karamshuk, N. Sastry, A. Secker, and J. Chandaria. ISP-friendly Peer-assisted On-demand Streaming of Long Duration Content in BBC iPlayer. In *Proceedings of INFOCOM*. IEEE, 2015.

[19] S. Kaune, R. C. Rumin, G. Tyson, A. Mauthe, C. Guerrero, and R. Steinmetz. Unraveling BitTorrent's File Unavailability: Measurements and Analysis. In *2010 IEEE Tenth International Conference on Peer-to-Peer Computing (P2P)*, pages 1–9. IEEE, 2010.

[20] A. Q. Lawey, T. E. El-Gorashi, and J. M. Elmirghani. Distributed Energy Efficient Clouds Over Core Networks. *Journal of Lightwave Technology*, 32(7):1261–1281, 2014.

[21] M. C. Lee, A. F. Molisch, N. Sastry, and A. Raman. Individual Preference Probability Modeling for Video Content in Wireless Caching Networks. In *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pages 1–7, Dec 2017.

[22] Z. Liu, Y. Ding, Y. Liu, and K. Ross. Peer-Assisted Distribution of User Generated Content. In *IEEE 12th International Conference on Peer-to-Peer Computing (P2P)*, pages 261–272. IEEE, 2012.

[23] Z. Lu, Y. Wang, and Y. R. Yang. An Analysis and Comparison of CDN-P2P-hybrid Content Delivery System and Model. *journal of communications*, 7(3):232–245, 2012.

[24] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan. A Power Benchmarking Framework for Network Devices. In *NETWORKING 2009*, pages 795–808. Springer, 2009.

[25] U. Mandal, M. F. Habib, S. Zhang, C. Lange, A. Gladisch, and B. Mukherjee. Adopting Hybrid CDNâĂŞP2P in IP-Over-WDM Networks: An Energy-Efficiency Perspective. *Journal of Optical Communications and Networking*, 6(3):303–314, 2014.

[26] D. S. Menasche, A. de A Rocha, B. Li, D. Towsley, and A. Venkataramani. Content Availability and Bundling in Swarming Systems. *Networking, IEEE/ACM Transactions on*, 21(2):580–593, 2013.

[27] S. Nedevschi, S. Ratnasamy, J. Padhye, and I. Reserch. Hot Data Centers vs. Cool Peers. In *HotPower*, 2008.

[28] G. Nencioni, N. Sastry, J. Chandaria, and J. Crowcroft. Understanding and Decreasing the Network Footprint of Catch-up TV. In *Proceedings of the 22nd international conference on World Wide Web*, pages 965–976. International World Wide Web Conferences Steering Committee, 2013.

[29] R. Peterson and E. G. Sirer. AntFarm: Efficient Content Distribution with Managed Swarms. In *NSDI*, volume 9, pages 107–122, 2009.

[30] F. Picconi and L. Massoulie. ISP-friendly Peer-to-Peer Live Streaming. Internet-draft, IETF Secretariat, March 2010.

[31] A. Raman, N. Sastry, A. Sathiaseelan, J. Chandaria, and A. Secker. Wi-Stitch: Content Delivery in Converged Edge Networks. *SIGCOMM Comput. Commun. Rev.*, 47(5):73–78, Oct. 2017.

[32] A. Raman, G. Tyson, and N. Sastry. Facebook (A)Live?: Are Live Social Broadcasts Really Broadcasts? In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pages 1491–1500. International World Wide Web Conferences Steering Committee, 2018.

[33] V. A. Siris and D. Kalyvas. Enhancing Mobile Data Offloading with Mobility Prediction and Prefetching. In *Proceedings of the seventh ACM international workshop on Mobility in the evolving internet architecture*, pages 17–22. ACM, 2012.

[34] V. Valancius, N. Laoutaris, L. Massoulié, C. Diot, and P. Rodriguez. Greening the internet with nano data centers. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 37–48. ACM, 2009.

[35] W. Vereecken, W. Van Heddeghem, M. Deruyck, B. Puype, B. Lannoo, W. Joseph, D. Colle, L. Martens, and P. Demeester. Power Consumption in Telecommunication Networks: Overview and Reduction Strategies. *Communications Magazine, IEEE*, 49(6):62–69, 2011.

[36] A. Vishwanath, K. Hinton, R. Ayre, and R. Tucker. Modeling Energy Consumption in High-Capacity Routers and Switches. *IEEE Journal on Selected Areas in Communications*, 32(8):1524–1532, Aug 2014.

[37] D. Wu, D. I. Arkhipov, T. Przepiorka, Y. Li, B. Guo, and Q. Liu. From Intermittent to Ubiquitous: Enhancing Mobile Access to Online Social Networks with Opportunistic Optimization. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3):114:1–114:32, Sept. 2017.

[38] G. Zhang, W. Liu, X. Hei, and W. Cheng. Unreeling Xunlei Kankan: Understanding Hybrid CDN-P2P Video-on-Demand Streaming. *IEEE Transactions on Multimedia*, 17(2):229–242, 2015.

[39] 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 *Proceedings of the IMC 2013*, pages 31–42. ACM, 2013.