How much can large-scale Video-On-Demand benefit from users' cooperation?'

How much can large-scale Video-On-Demand benefit from users' cooperation?

Delia Ciullo, Valentina Martina, Michele Garetto, Emilio Leonardi(*)

***Abstract*—We propose an analytical framework to tightly characterize the scaling laws for the additional bandwidth that servers must supply to guarantee perfect service in peer-assisted Video-on-Demand systems, taking into account essential aspects such as peer churn, bandwidth heterogeneity, and Zipf-like video popularity. Our results reveal that the catalog size and the content popularity distribution have a huge effect on the system performance. We show that users' cooperation can effectively reduce the servers' burden for a wide range of system parameters, confirming to be an attractive solution to limit the costs incurred by content providers as the system scales to large populations of users.**

I. Introduction and previous work

According to Cisco [1], by the end of 2016 the sum of all forms of Internet video (TV, Video-on-Demand, P2P) will be approximately 86% of global consumer traffic. In particular, the traffic component due to Video-on-Demand is expected to triple from 2011 to 2016, reaching the equivalent of 4 billion DVDs per month.

Increasing traffic volumes force video providers to continuously upgrade the Content Delivery Network (CDN) infrastructure that feeds the contents to local ISPs. To partially alleviate this burden, a recent trend of VoD providers is to exploit cloud services, which permit fine-grained resource reservation [2]. As an example, in 2010 Netflix decided to migrate its infrastructure into the Amazon EC2 cloud, as it could not build data centers fast enough to keep pace with growing demand.

However, any solution based on CDNs has severe limitations in terms of scalability. CDNs can significantly reduce the traffic in the Internet core and improve the user-perceived performance (*e.g.*, by reducing the latency) by "moving" contents close to the users. Nevertheless, the aggregate resources required at data centers (bandwidth/storage/processing), and the corresponding costs incurred by content providers, inevitably scale linearly with the user demand and data volume.

The only scalable solution proposed so far is to exploit the peer-to-peer paradigm, according to which users contribute their resources (bandwidth/storage/processing) to the system while they use it [3], [4]. Although the peer-assisted approach is an attractive solution to the scalability problem, and it has already been experimented in several applications [5], it brings with it several issues which tend to discourage its adoption by many content providers: the unpredictable nature of users' cooperation, the added complexity on the control plane due to signalling and chunk scheduling, and the need to provide incentive mechanisms to the users [6].

Streaming architectures which primarily rely on users' cooperation can hardly guarantee the strict quality-of-service requirements of online video, where a steady download rate no smaller than the video playback rate is necessary for a smooth watching experience, and any interruption tends to be very annoying to the user [7].

For this reasons, we argue that peer-assisted architectures should be supported by properly dimensioned CDNs (or cloud services) that intervene whenever the resources provided by users are not enough to satisfy the current demand. In our theoretical work, we are specifically interested in characterizing the additional bandwidth that servers must supply to guarantee ideal service to all users (*i.e.*, requests are immediately satisfied and videos can be watched without interruptions till the end). Our main contribution is a stochastic analytical framework that allows to derive general upper and lower bounds to the bandwidth requested from the servers in a peer-assisted VoD system, capturing essential aspects such as peer churn, bandwidth heterogeneity, and Zipf-like video popularity. Our analysis permits to tightly characterize the system performance as the number of users (and the number of available videos) grows large, and thus assess the scalability of large-scale VoD exploiting users' cooperation.

In our previous work [8] we have considered the case of a single-video, providing for the first time an asymptotic characterization of the servers' bandwidth as the number of watchers increases. Here we extend the analysis to a multi-video system, in which users can browse a catalog of available contents, and asynchronously issue requests to watch videos.

Our main contribution is a precise definition of the conditions (related to physical system parameters such as the growth rate of the catalog size, the Zipf's exponent of video popularity, videos' characteristics and user behavior) under which the additional bandwidth requested from the servers asymptotically goes to zero as the size of the system grows large. When such conditions are not met, we provide the asymptotic laws for the required servers' bandwidth.

We consider both the case in which users can only assist the distribution of the last video they have selected (we call this the *passive* system, because the utilization of peer resources is tied to the video popularity distribution, which is not under the system control), and the general case in which users can assist the distribution of any video (we call this the *active* system, also referred to in the literature as *universal streaming*). For the *active* system we also devise the resource allocation strategies that permit to achieve the optimal theoretical performance.

We emphasize that a full exploitation of peers' upload bandwidth is not a trivial task in the presence of high degrees of peer churn (*i.e.*, when users tend to abandon the system after watching a few videos), in consideration of the obvious fact that users can only upload data that they have previously downloaded. For the same reason unpopular videos, which tend to be scarcely replicated among peers, can pose a significant stress on the

(*) D. Ciullo is with INRIA Sophia Antipolis, France; V. Martina and E. Leonardi are with Dipartimento di Elettronica, Politecnico di Torino, Italy; M. Garetto is with Dipartimento di Informatica, Università di Torino, Italy.

system. Hence another important contribution of our work is the definition of suitable strategies to mitigate the joint impact of peer churn and heterogeneous video popularity.

Universal streaming architectures have been analytically studied in [9], where authors develop queueing network models to describe multi-channel live streaming systems incorporating peer churn, bandwidth heterogeneity, and Zipf-like popularity. We remark that VoD systems are different from live streaming systems in which users join the distribution of a given TV channel at random points in time, but peers connected to the same channel watch the content almost synchronously. In VoD, a given video is watched asynchronously by users, and downloading peers can only help peers who have started the download later on in time (sequential delivery). Moreover, asymptotic results in [9] are restricted to the case of two values of peer upload bandwidth (low and high), and require finding the solution (if any) to a set of linear equations. In contrast to [9], we consider VoD systems, and obtain a simpler characterization of the asymptotic system performance for general upload bandwidth distribution.

The first mathematical formulation of the server bandwidth needed by a VoD system based on sequential delivery appeared in [4], in which authors resort to a Monte Carlo approach to get basic insights into the system behavior (like surplus and deficit modes). The same formulation has been considered in [10], where authors explore by simulation the effectiveness of different replication strategies to minimize the server load in the slightly surplus mode, as well as distributed replacement algorithms to achieve it.

An interesting implementation of the kind of systems considered in our work is Xunlei [11], a download acceleration application that is becoming enormously popular in China. Xunley combines both peer-assisted and server-assisted techniques, letting users download portions of the requested contents from other peers while also downloading portions from independent servers. Recently, the Xunlei network started also a peer-assisted VoD service (Kankan), which generated massive-scale swarms.

II. System assumptions

A. Service specification and users cooperation

We model a VoD system where users run applications that allow them to browse an online catalog of videos. When a user selects a video, we assume that the request is immediately satisfied and the selected video can be watched uninterruptedly till the end, *i.e.*, the system is able to steadily provide to the user a data flow greater than or equal to the video playback rate. We consider that users watch at most one video at a time.

We assume that the system catalog contains K different videos. Video k ($1 \leq k \leq K$) is characterized by: its size $l_k \in [l_{\min}, l_{\max}]$, expressed in bytes; a selection probability p_k, which is the probability that a user selects video k among all videos in the catalog; a minimum playback rate. We assume that video k is downloaded at constant rate greater than or equal to the minimum playback rate. Specifically, we denote by d_k the download rate, where $d_k \in [d_{\min}, d_{\max}]$ (in bytes/s).

Users contribute their upload bandwidth to the video distribution: they can retrieve part of a requested video (or even the entire video) from other users, saving servers resources.

We model the amount of upload bandwidth contributed at a given time by a user by a random variable U with cumulative distribution function F_U and mean $\overline{U}$, in this way we take into account effects related to Internet access heterogeneity and cross traffic fluctuations. The random variables U's denoting the upload bandwidths of the users are assumed to be i.i.d.

Users contribute to the system also a limited amount of storage capacity. The exact amount of buffer space available at each user is not important in our analysis. As a minimum requirement, our schemes assume that users can store at least one whole video in addition to the one currently played out.

B. User dynamics

Users join the system when they request the first video. We denote by λ^u the arrival rate of new users. While they are in the system, users can be in two states: {*contributing, sleeping*}. The *contributing* state is defined as the state in which a user is contributing its upload bandwidth to the system. In the *contributing* state, a user can download (and watch) video contents. Notice that a user can be contributing its upload bandwidth even if it is not currently downloading/watching any video, but simply because it keeps its VoD application up and running.

During the *sleeping* phase, the user's application is not running, hence it is neither downloading nor uploading data. We assume that users download the entire requested videos (aborted downloads could be easily included in our model but we have preferred not to do so for simplicity). Note that, since a video is retrieved at constant rate, its download time, $\tau_k = l_k/d_k$, is a deterministic attribute of video k, taking values in range $[\tau_{\min} = l_{\min}/d_{\max}, \tau_{\max} = l_{\max}/d_{\min}]$. After completing a download, users remain in the *contributing* state for a random amount of time T_{seed} with mean $\overline{T}_{\text{seed}}$ (part of this time can be spent finishing to watch the video, if the download rate is larger than the playback rate). Then, they transit to the *sleeping* state, where they stay for a random amount of time of mean $\overline{T}_{\text{sleep}}$. Users can choose to abandon the system (*i.e.*, to stop the VoD application and never open it again) after watching just a single video. We assume that, after watching a video, each user independently decides to leave the system with probability p_{out}. It follows that the number of videos requested by a user is geometrically distributed with mean $m = 1/p_{\text{out}}$. Moreover, the average time spent by a user in the system can be computed as $\overline{T} = m \cdot (\sum_{k=1}^{K} p_k \tau_k + \overline{T}_{\text{seed}} + \overline{T}_{\text{sleep}})$.

From the above assumptions, and the fact that the system provides guaranteed service, the set of videos requested by a user, the total time spent by a user in the system, as well as the amounts of time spent by a user in the *contributing/sleeping* states are independent from user to user.

C. System scaling

Our goal is to asymptotically characterize the average additional bandwidth $\overline{S}$ that servers must supply to guarantee perfect service to all users, as the system grows large. Let n be the average number of users in the system. By Little's law, we have $n = \lambda^u \overline{T}$. Note that $\overline{T}$ is a constant, hence our asymptotic analysis for increasing number of users is performed by letting λ^u (and thus n) go to infinite.

Since the catalog size is expected to grow, just like the number of users, we consider that the number K of videos available in

the catalog is tied to the number of users, according to the law $K = \Theta(n^\beta)$, with $\beta \leq 1$[1].

As the system grows, new videos are made available to the users. We assume that the characteristics of new videos inserted into the catalog, in terms of file size l_k and download rate d_k, are random. Hence l_k and d_k should be regarded as instances of i.i.d. random variables L_k and D_k, respectively, with assigned distributions (possibly correlated). Recall from Section II-A that we (reasonably) assume that the distributions of L_k, D_k have finite support independent of n.

D. Content popularity

To specify the selection probabilities of videos, we need to model the relative popularity of the videos in the catalog. For this, we adopt the standard Zipf's law, which has been frequently observed in traffic measurements and widely adopted in performance evaluation studies [9], [12]. More specifically, having sorted the videos in decreasing order of popularity, a request is directed to video k with probability

$$p_k \triangleq \frac{H(K)}{k^\alpha} \quad , 1 \leq k \leq K \tag{1}$$

where α is the Zipf's law exponent, and $H(K) \triangleq (\sum_{i=1}^{K} i^{-\alpha})^{-1}$ is a normalization constant. Depending on the exponent α, we have:

$$H(K) \quad = \quad \begin{cases} \Theta(1) & \text{if } \alpha > 1 \\ \Theta(\log K) & \text{if } \alpha = 1 \\ \Theta(K^{\alpha-1}) & \text{if } \alpha < 1 \end{cases} \tag{2}$$

Let Λ be the aggregate rate at which users request videos. By construction $\Lambda = \lambda^u m$. The rate at which a specific video k is requested is $\lambda_k = \Lambda p_k$.

E. System load

For a given system catalog, *i.e.*, for given video characteristics $\{d_k\}_k$ and $\{l_k\}_k$, we can compute a fundamental quantity γ characterizing the global system load (i.e., the load induced by all videos):

$$\gamma \triangleq \frac{\sum_{k=1}^{K} p_k \, l_k}{\sum_{k=1}^{K} p_k \, \overline{U} \, (\tau_k + \overline{T}_{\text{seed}})} \tag{3}$$

Indeed, consider a large time interval Δ. During this time interval, a video k will be requested on average $\lambda_k \Delta$ times. Each request for video k has a double effect on the system: it requires an amount of bytes l_k to be downloaded; it lets the requesting user potentially to upload an average amount of data $\overline{U} \, (\tau_k + \overline{T}_{\text{seed}})$. The ratio between the average amount of downloaded data and the average amount of uploaded data during interval Δ, for $\Delta \to \infty$, leads to the expression in (3).

We remark that (3) holds for both *passive* and *active* systems introduced in Section I. However, in the case of *active* systems it does not account for the additional data that users might be instructed to download by the system (data bundling). The effect of bundling on the system load will be considered later. Borrowing the terminology adopted in previous work [3], [10] we say[2] that the system operates in *deficit* mode if $\gamma > 1$, and in *surplus* mode if $\gamma < 1$.

[1]We leave to future work the case $\beta > 1$.

[2]In this paper we do not consider the special case $\gamma = 1$.

TABLE I

Symbol	Definition
λ^u	user's arrival rate
$\overline{T}$	average time spent by users in the system
n	average number of users
K	catalog size (number of available videos)
β	scaling exponent of $K = n^\beta$
Λ	aggregate video request rate
λ_k	request rate of video k
d_k	download rate of video k
τ_k	download time of video k
$\overline{U}$	average user upload bandwidth
$\overline{T}_{\text{seed}}$	average time spent in the *contributing* state after downloading a video
$\overline{N}_{d,k}$	average number of users downloading video k
$\overline{N}_{\text{seed},k}$	average number of *seeds* for video k
$\overline{S}$	average bandwidth requested from the servers
γ	system load
γ_k	load associated to video k
m	average number of videos requested by a user

We emphasize that, since video characteristics are random, γ should be itself interpreted as an instance of a random variable Γ obtained de-conditioning (3) with respect to $\{d_k\}$ and $\{l_k\}$.

We will also use a video-specific notion of load, denoted by γ_k, and its corresponding random variable Γ_k:

$$\gamma_k \triangleq \frac{d_k \, \tau_k}{\overline{U} \, (\tau_k + \overline{T}_{\text{seed}})} \tag{4}$$

We observe that γ_k would coincide with γ if all γ_k were equal. With abuse of language, we say that a video is in *deficit* mode if $\gamma_k > 1$, and in *surplus* mode if $\gamma_k < 1$.

Table I summarizes the notation introduced so far.

III. Summary of results

First we observe that, in the worst possible case, the servers have to transmit at rate $d_{\max}$ to all downloading users. It follows that a trivial upper bound to the bandwidth requested from the servers is $\overline{S} = O(n)$. A trivial lower bound is $\overline{S} \geq 0$.

For the *passive* system, we obtain the following results. If the probability to include in the catalog a video with load $\gamma_k > 1$ is greater than zero, *i.e.*, $\mathbb{P}(\Gamma_k > 1) > 0$, we have $\overline{S} = \Theta(n)$. If, instead, there exists an arbitrarily small constant σ such that $\mathbb{P}(\Gamma_k < 1 - \sigma) = 1$, we obtain the asymptotic upper bounds reported in the second column of Table II, which depend on the Zipf's exponent α and the catalog growth rate exponent β.

For the *active* system, we obtain the following results. If $\Gamma > 1$ with non vanishing probability as the system size increases, we have $\overline{S} = \Theta(n)$. If $\Gamma < 1 - \sigma$ (w.h.p.)[3], for some $\sigma > 0$, we obtain the asymptotic upper bounds reported in the third column of Table II, which depend on the Zipf's exponent α and the catalog growth rate exponent β, while δ is an arbitrarily small positive number.

The fourth column of Table II reports corresponding lower bounds for $\overline{S}$, which are valid also for the extreme case in which the user upload bandwidth is arbitrarily large.

Our results for the *active* system provide the following fundamental insights: if $\beta < 1$, *i.e.*, if the number of contents in the system scales sub-linearly with respect to the average number of users, an active system operating (globally) in surplus mode can asymptotically eliminate the need of additional bandwidth from the servers (*i.e.*, $\overline{S}$ tends to zero as the number of users

[3]With high probability, i.e., with probability that tends to 1 as $n \to \infty$.

TABLE II
AVERAGE BANDWIDTH REQUESTED FROM THE SERVERS, $\overline{S}$

Conditions	Upper bound: *Passive* system $\mathbb{P}(\Gamma_k < 1-\sigma) = 1$	Upper bound: *Active* system $\mathbb{P}(\Gamma < 1-\sigma) \to 1$	Lower bound
$(\alpha \le 1 \wedge \beta < 1) \vee (\alpha > 1, \beta < 1/\alpha)$	$o(1)$	$o(1)$	0
$\alpha > 1 \wedge 1/\alpha < \beta < 1$	$O(n^{1/\alpha})$	$o(1)$	0
$\alpha > 1 \wedge \beta = 1$	$O(n^{1/\alpha})$	$O\left(n^{2-\alpha}(\log n)^{\frac{\alpha-1}{1-\delta}}\right)$	$\Omega(n^{2-\alpha})$
$\alpha \le 1 \wedge \beta = 1$	$O(n)$	$O(n)$	$\Theta(n)$

increases), for any value of α. This can be done even under the sequential delivery scheme (*i.e.*, when downloading users can only help future downloaders of the same file). We remark that the only requirement to achieve this desirable behavior is that the global system load is smaller than one, which does not imply that all videos are individually in the surplus mode.

If, instead, $\beta = 1$, (*i.e.*, when the number of contents in the system scales as fast as the number of users)[4], the exploitation of users' cooperation is more difficult and depends on the Zipf's exponent: for $\alpha \le 1$ we cannot do any better than the worst case $\overline{S} = \Theta(n)$. For $\alpha > 1$ (and $\Gamma < 1 - \sigma$), we approximately need $n^{2-\alpha}$ additional servers' bandwidth (notice that in this case upper and lower bounds differ only by a poly-log term), which unfortunately goes to infinite for any $\alpha < 2$. This is essentially due to the fact that, for $\beta = 1$, there are too many contents available in the system (whose aggregate data volume becomes comparable to the total buffer space available at users). In this case, contents cannot be distributed/replicated at peers in such a way that the distribution of all of them can be effectively assisted by the users, considering also the effect of peer churn.

Passive systems perform, obviously, worse than active systems. First of all, they can lead to something better than $\overline{S} = \Theta(n)$ only when all videos are in surplus mode, which is a rather restrictive condition. Nevertheless, if this condition is satisfied, for $\beta < 1$ we still obtain that $\overline{S} = o(1)$, provided that $\alpha \le 1$ or $\beta < 1/\alpha$. Instead, the servers' bandwidth goes to infinite as $n^{1/\alpha}$ for $\beta > 1/\alpha$, and the same occurs if $\beta = 1$ (of course without exceeding the worst case $\overline{S} = \Theta(n)$).

IV. PASSIVE SYSTEM

We start considering the *passive* system, in which users are constrained to assist only the distribution of the last selected video. This means that, after requesting a video, they can only download/upload data belonging to the selected video (until they request a new content from the catalog). A passive system is conceptually simple to implement and manage, since swarms of different videos are decoupled, and can be controlled independently of each other.

A. Preliminaries

We can describe the dynamics of users in the system by the open queueing network illustrated in Fig. 1. We consider a separate queue for all users downloading the same video. When the download is complete, users who keep the application running continue contributing their uploading bandwidth to the system, transiting to queues arranged in the second column of the network. Users who stop the application transit to the *sleeping* state, represented by a single queue on the right hand side.

[4]Our results could be extended to the case $\beta > 1$, reaching identical conclusions.

Lemma 1: At any time, the number of users who are downloading a given video, the number of users who remain in the *contributing* state after downloading a video, and the number of users in the *sleeping* state, follow independent Poisson distributions.

Proof: The dynamics of users in the open queueing network (in terms of transitions among the queues and sojourn times at queues) are decoupled, since users behave independently of each other[5]. The resulting queueing network admits a product-form solution by the BCMP theorem. Since all queues have infinite servers, the numbers of users in the queues follow independent Poisson distributions. ■

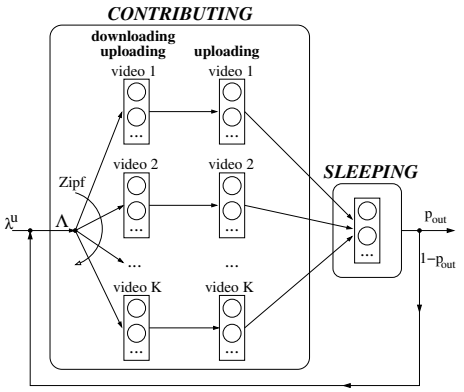

Fig. 1. Open network of $\cdot/G/\infty$ queues modeling users' dynamics.

Let $N(t)$ be the total number of users in the system at time t. Note that $N(t)$ is itself Poisson distributed, with mean $n = \lambda^u \overline{T}$. We denote by $\overline{N}_{d,k} = \lambda_k \tau_k$ the average number of users downloading file k, and by $\overline{N}_{\text{seed},k} = \lambda_k \overline{T}_{\text{seed}}$ the average number of users remaining in the *contributing* state after downloading file k. In a *passive* system, $\overline{N}_{\text{seed},k}$ represents also the average number of users acting as seeds for video k.

B. Asymptotic results for single video system

Before considering the bandwidth requested from the servers to support the distribution of all videos available in the catalog, we analyze the simple case in which there is just one video (*i.e.*, $K = 1$), whose request rate λ tends to infinite. Notice that in this case γ in (3) equals γ_1 in (4).

The following theorem, which is unfortunately a bit technical and strongly based on our previous work [8], characterizes how the servers' bandwidth $\overline{S}$ scales with λ:

Theorem 1: Assume the following properties hold for U: i) $\overline{U} > 0$, ii) $\mathbb{E}[e^{\theta U}]$ is finite in a neighborhood of the origin, iii) $F_U(w) > 0$ for every $w > 0$. The average bandwidth requested from the servers, $\overline{S}$, satisfies the following asymptotic bound as $\lambda \to \infty$: if $\gamma < 1$, for any $\delta \in (0, 1)$,

$$\overline{S} \le \begin{cases} 2\lambda^{\delta} e^{-C_1 \lambda^{1-\delta}} & \text{if } d > \overline{U} \\ C_3 e^{-C_2 \lambda} & \text{if } d \le \overline{U}, \end{cases} \tag{5}$$

with $C_1 = \frac{1}{2}\tau \frac{d(1-\gamma)}{\gamma}$, $C_2 = \left(\frac{d}{\overline{U}\gamma} - 1\right)\tau(1 - e^{-\theta^* d})$, $C_3 = (d + 1/\theta^*)e^{\theta^* d}$, where θ^* is the only strictly positive

[5]Notice that here we are not considering as part of user dynamics the data downloaded/uploaded by a user, which obviously depend on which videos the other users have requested.

solution of the equation $\mathbb{E}[e^{\theta(d-U)}] = 1$. Furthermore (5) holds uniformly with respect to U, d and γ as long as $\gamma < 1 - \sigma$ for any $\sigma > 0$. If, on the other hand, $\gamma > 1$, the average bandwidth requested from the servers grows linearly with the number of users, namely, $\overline{S} = \Theta(\lambda)$.

A detailed proof of Theorem 1 is reported in Appendix A. We emphasize that Theorem 1 is not a straightforward extension of the results in [8], where we have only shown that $\overline{S} \to 0$ as $\lambda \to \infty$ (provided that $\gamma < 1$). Indeed, Theorem 1 characterizes also *how* $\overline{S}$ scales with λ, providing a basic building block of our analysis.

Considering the constants specified in Theorem 1, we observe that C_1 is insensitive to the distribution of U, and it does not depend explicitly from $\overline{U}$. Actually, its dependency from $\overline{U}$ is mediated by γ. As consequence, the expression of the bound for $d > \overline{U}$ is robust to the distribution of U and its mean, provided that $\gamma < 1-\sigma$. Instead, C_2 and C_3 are sensitive to the distribution of U, through the quantity θ^*. For this reason, the upper bound on $\overline{S}$ for $d \leq \overline{U}$ is more delicate. In particular, note that if θ^* becomes arbitrarily small or large the bound $C_3 e^{-C_2 \lambda}$ becomes arbitrarily weak.

Corollary 1: Under the same assumptions on U of Theorem 1, if $\gamma < 1 - \sigma$ for some $\sigma > 0$, uniformly over $\overline{U}$, d and γ a $\lambda_0 > 0$ can be found such that:

$$\overline{S} \leq 2\lambda^{\delta} e^{-C_1(d,\tau,\gamma')\lambda^{1-\delta}} \ \forall \delta \in (0,1) \text{ and } \lambda > \lambda_0 \qquad (6)$$

with $\gamma' = \gamma(1 + \sigma/2)$.

Proof: Exploiting the definition of θ^*, we can derive the following lower and upper bound for θ^*: $\theta^* > \frac{2E[U-d]}{\mathbb{E}[(U-d)^2]}$ and $\theta^* < \sup_{x<d} -\frac{\log \mathbb{P}(U<d-x)}{x}$, which guarantees that θ^* can not be arbitrarily small or large when d is sufficiently smaller than $\overline{U}$, let us say when $d(1 + \sigma/2) \leq \overline{U}$. Therefore, whenever $\overline{U} > d(1+\sigma/2)$ we can jointly lower bound C_2 by a positive constant, and upper bound C_3 by a constant. It follows that $C_3 e^{-C_2\lambda} = o(\lambda^{\delta} e^{-C_1 \lambda^{1-\delta}})$, as $\lambda \to \infty$, $\forall \delta \in (0,1)$.

When $\overline{U} \to d$ from the right, $\theta^* \to 0$ and the bound in Theorem 1 becomes arbitrarily weak (it tends to infinite). To overcome this problem, we exploit the following trick to get a useful bound also for $d \leq \overline{U} < d(1 + \sigma/2)$: we assume that peers contribute only a fraction $\frac{2}{2+\sigma}$ of their upload bandwidth to the video distribution. By so doing, we waste a small fraction of the peers upload bandwidth, increasing the video load to $\gamma' = \gamma(1 + \sigma/2) < 1$. Moreover, we end up with a system in which the average effective upload bandwidth becomes smaller than d. For this system, we can bound $\overline{S}$ by the expression valid for $d > \overline{U}$, obtaining a bound clearly valid also for the case in which the peer upload bandwidth is fully utilized. ■

The upper bound stated in Theorem 1 (and Corollary 1) is valid for $\lambda \to \infty$. If we want to apply this result to a multi-video system, we must be careful that the video request rates λ_k might not all tend to infinite as $n \to \infty$. In general, we can divide the video catalog into two portions: the *hottest* portion of the catalog comprises videos whose request rate tends to infinite as $n \to \infty$; the *coldest* portion of the catalog (which could be empty) comprises videos whose request rate remains constant or eventually drops to zero as $n \to \infty$.

For videos in the *coldest* portion, we need a different bound, since for them we cannot apply the result in Corollary 1. Notice that, for any cold video, peer assistance is rather ineffective (in a passive system), because it is even possible that at a given time there are no seeds supporting its distribution (*e.g.*, they might be all sleeping). A very crude bound that we can use for cold videos is based on the pessimistic assumption that the entire bandwidth necessary to sustain their downloads is provided by servers (*i.e.*, neglecting the contribution of seeds and simultaneously downloading users):

Lemma 2: A universal upper bound to the bandwidth requested from the servers is: $\overline{S} \leq d\tau\lambda$.

Although this bound may appear particularly coarse, it captures well (in order sense) the impact that cold videos have on the aggregate bandwidth requested from the servers, as we will see. Combining Corollary 1 and Lemma 2, we get:

Corollary 2: Under the assumptions on U of Theorem 1, if $\gamma < 1 - \sigma$, for some $\sigma > 0$, then uniformly over $\overline{U}$, d and γ a $\lambda_0 > 0$ can be found such that

$$\overline{S} \leq \begin{cases} d\tau\lambda & \text{if } \lambda < \lambda_0 \\ 2\lambda^{\delta} e^{-C_1(d,\tau,\gamma')\lambda^{1-\delta}} \ \forall \delta \in (0,1) & \text{if } \lambda \geq \lambda_0 \end{cases} \qquad (7)$$

with $\gamma' = \gamma(1 + \sigma/2)$.

C. Asymptotic results for multi-video system

Corollary 2 can be readily exploited to compute the aggregate bandwidth requested from the servers in the case of multi-video systems. Indeed, we basically have to add up the contributions of individual videos to the bandwidth requested from the servers. We obtain:

Theorem 2: Under the same assumptions on U of Theorem 1, if there exists an arbitrarily small constant $\sigma > 0$ such that $\mathbb{P}(\Gamma_k < 1 - \sigma) = 1$, than the average bandwidth $\overline{S}$ requested from the servers satisfies the following asymptotic bound w.h.p. as the number of users n tends to ∞:

$$\overline{S} = \begin{cases} O(n^{1/\alpha}) & \text{if } \alpha > 1, 1/\alpha < \beta \leq 1 \\ o(1) & \text{if } (\alpha \leq 1, \beta < 1) \vee (\alpha > 1, \beta < 1/\alpha) \\ O(n) & \text{if } \alpha \leq 1, \beta = 1 \end{cases}$$

If, instead, $\mathbb{P}(\Gamma_k \geq 1) > 0$, then $\overline{S} = \Theta(n)$ w.h.p., $\forall\, \beta > 0$.
A detailed proof is reported in Appendix B.

Remarks. We emphasize that when all videos are in the surplus mode, the dominant contribution to the bandwidth requested from the servers is always due to the coldest portion of the video catalog. In particular, when $\alpha > 1$, $1/\alpha < \beta < 1$, the scaling law of $\overline{S}$ is determined by videos whose request rate either remains constant or decays to zero.

Although a passive system is conceptually simple to implement and manage, it is a very rigid (and potentially suboptimal) scheme, since users are constrained to devote their entire upload bandwidth to the last requested video, and by so doing their resources might not be fully utilized. In the next section, guided by the insights gained from the analysis of passive systems, we will investigate the performance achievable by active systems.

V. Active systems

In active systems, users can be instructed to essentially download/upload data belonging to arbitrary videos, with the obvious constraints that: i) they must at least download (at constant rate) the videos that they want to watch; ii) they can upload only data previously downloaded. In particular, users can download/upload chunks or stripes belonging to videos they have not requested (data bundling). However, we will not consider the extreme case

in which chunks/stripes can be made arbitrarily small (fluid limit), *i.e.*, chunks/stripes whose size asymptotically goes to zero, because this is not implementable in practice.

We will show that, even with this restriction (*i.e.*, the size of chunks/stripes can not go to zero) we can devise efficient active strategies that can overcome the fundamental limitations of passive systems. In particular, we need to solve two orthogonal problems: i) the possible presence of videos in deficit mode, which prevents any passive system to scale (we call this the *load balancing* problem); the possible presence of cold videos, which are especially detrimental to the system (we call this the *catalog warming* problem). For each problem, we will present more than one solution, reporting the main results and the basic intuition about how they work. We anticipate that, once we solve both problems above, the computation of the resulting system performance will be an easy task. Due to the lack of space, missing proofs can be found in [13].

A. Load balancing by seed reallocation

The goal of video equalization is to make the loads induced by individual videos equal to the global system load (3). The simplest approach to redistribute the peer upload resources to achieve this goal is to remove the constraint that peers must act as seed only for the last downloaded video. In this way, we can allocate extra seeds to those videos having $\gamma_k > \gamma$ in the passive system. Although the approach is simple (it does not require any chunk/stripe bundling), the performance of this strategy is clearly limited by the fact that users download only a finite number of videos m before leaving the system, hence they cannot act as seed for arbitrary videos.

We propose a seed reallocation strategy that works as follows: i) all videos are downloaded at the same speed $d_{\max}$; this way we decrease the download time (that becomes $\tau'_k = l_k/d_{\max}$) and increase the average time during which peers may act as seed after downloading video k (we denote this quantity by $\overline{T}_{\text{cont},k} = \overline{T}_{\text{seed}} + \tau_k - \tau'_k$). ii) peers acting as seeds are divided into $K+1$ categories: seeds assigned to a specific video and unassigned seeds. Seeds assigned to video k act as seed for video k for all their residual lifetime in the system; unassigned seeds, instead, act as seeds for the last downloaded video. Every fresh new peer joining the system is initially unassigned. An unassigned peer, after downloading video k, is assigned to video k with probability q_k, while it remains unassigned with probability $1 - q_k$.

Theorem 3: Given $\{d_k\}_k$, $\{l_k\}_k$, the proposed seed reallocation strategy guarantees perfect load balancing (by properly selecting probabilities q_k), iff $\gamma < 1$, and the following condition on τ'_k, $\overline{T}_{\text{cont},k}$ is satisfied:

$$\max_k \left[\tau'_k \cdot \max_h \left[\frac{\overline{T}_{\text{cont},h}}{\tau'_h} \right] - \overline{T}_{\text{cont},k} \right] \leq (m-1)\overline{T}_{\text{cont}} \quad (8)$$

where $\overline{T}_{\text{cont}} = \sum_k p_k \overline{T}_{\text{cont},k}$.

De-conditioning with respect to d_k and l_k we obtain that a perfect balance of video loads is feasible w.h.p. iff

$$l_{\max}\left(\frac{\overline{T}_{\text{seed}}}{l_{\min}} - \frac{\overline{T}_{\text{seed}}}{l_{\max}} + \frac{1}{d_{\min}} - \frac{1}{d_{\max}} \right) \leq (m-1) \mathop{\mathbb{E}}_{L_k, D_k} [\overline{T}_{\text{cont}}]$$

Note that, if users stayed indefinitely in the system, they would sooner or later download any video that requires additional seeds, hence by properly setting probabilities q_k we would surely be able to equalize the loads. Theorem 3 provides the sufficient and necessary condition on the average number of videos m downloaded by a user (which is proportional to average residence time in the system) such that perfect load balancing is still possible. Previous approach can be further boosted by allowing view-upload decoupling, *i.e.*, by letting users assigned to video k to act as seed for it also while they are downloading a different video. The resulting condition on m is obtained by replacing $\overline{T}_{\text{cont}}$ with $\overline{T}_{\text{cont}} + \overline{\tau}'$, where $\overline{\tau}' \triangleq \sum p_k l_k / d_{\max}$, on the right hand side of (8).

B. Load balancing by stripe bundling

This technique is based on the following idea: each video is divided into M stripes (substreams), which have to be downloaded in parallel by a user requesting the video (the distribution of each stripe can be assisted by a different set of peers), and reassembled by the decoder. Users who are downloading a video in surplus mode are forced to download also one stripe of a video in deficit mode, and devote a fraction of their upload bandwidth (actually, all of their excess bandwidth with respect to the target average system load) to the bundled stripe. The following theorem guarantees that, by making M large enough (but not infinite), we can approximately balance the loads bringing all videos in surplus mode.

Theorem 4: For any value γ' such that $\gamma < \gamma' < 1$, there exists a value $M^* < \infty$ for the number of stripes such that for all $M > M^*$ a stripe bundling scheme can be found that brings the system to operate at global load smaller than γ'. At the same time, the load associated to each video becomes smaller than or equal to γ' (the same holds considering the load induced by individual stripes).

We limit ourselves to providing an intuitive understanding of why this strategy turns out to be very effective to balance the video loads while minimally increasing the global system load. Indeed, while on the one hand some users (those requesting a video in surplus mode) have to download additional unwanted data (but this additional amount of data, corresponding to a single stripe, can be made smaller and smaller by increasing M), on the other hand these users can exploit all of their excess upload bandwidth to assist the distribution of the bundled stripe, typically retransmitting many copies of it to other peers before leaving the system, with an obvious gain in terms of system performance. This technique is more complex to implement than the previous one based only on seed reallocation. However, it has the great advantage that it does not require any additional condition on the system parameters. In particular, it works also in the extreme case in which users leave the system after downloading just one video ($m = 1$).

C. Catalog warming by video bundling

While analysing the case of a passive system, we learnt that cold videos (videos whose request rate does not increase with n) are responsible for the dominant component of the bandwidth requested from the servers. Hence, if we could artificially increase the request rate of cold videos, we would expect to get a significant reduction of $\overline{S}$. Now, it turns out that we do not need to warm up the coldest portion of the catalog too much: optimal performance is already achieved when the request rate of videos go to infinite at least as fast as a poly-log function, *i.e.*, when $\lambda_k = \Omega((\log n)^z)$, $\forall k$ (actually, this is needed only for a

'critical' portion of the catalog) where z is a suitable constant. The amount of data bundling necessary to achieve this goal is rather small, hence it is possible to warm the catalog up enough, while at the same time increasing the global system load to a value γ' such that

$$\gamma' - \gamma = \Delta\gamma \to 0, \quad \text{for } n \to \infty \tag{9}$$

The simplest approach to achieve this goal is to make some peers in the *contributing* state to download entire (unrequested) videos while they are not downloading any other content. This mechanism does not require any video chunking/striping, and can be superposed to the load balancing strategy described in Section V-A. In essence, the strategy works as follows. Let λ'_k be the target new rate at which video k should be downloaded in the active system. Peers who have just finished to download a hot video $h \leq K_0$ (for a specific constant K_0), without been assigned to it by the seed reallocation strategy, are induced to start the download of a cold video $k > K_0$ with probability $p'_k = \frac{(\lambda'_k - \lambda_k)}{\sum_{h \leq K_0} \lambda^u p_h (1-q_h)}$ (note that, by construction, $p_d = \sum_{k>K_0} p'_k \to 0$ as $n \to \infty$). If the download of the bundled video is interrupted because the peer goes into the sleeping state, the download is promptly resumed as soon as the peer restarts contributing to the system without concurrently downloading any other video (they can not be assigned to any video they possibly download in the meanwhile). Since $p_d \to 0$, the negative effect that this strategy has on the load induced by hot videos (those videos whose request rate is not increased, and from which the mechanism subtracts some seeds) becomes negligible for $n \to \infty$. Our strategy has a potential effect also on the load of videos whose request rate is artificially increased. However, it guarantees that the new load γ'_k of such videos is maintained less than 1, provided that the average upload bandwidth of adjoint seeds exceeds the average bandwidth consumed to download them, *i.e.*, $d_k \min(m\overline{T}_{\text{cont}}, l_k/d_k) < \overline{U} m \overline{T}_{\text{cont}}$. When this condition is met with probability 1, *i.e.*,

$$\mathbb{P}\left(\frac{d_{\max} \min(m\overline{T}_{\text{cont}}, l_{\max}/d_{\max})}{m\overline{UT}_{\text{cont}}} > 1 \right) = 1, \tag{10}$$

the above scheme can be effectively employed, in sufficiently large systems, without bringing any video in deficit mode.

D. Catalog warming by chunk bundling

The previous technique imposes again a constraint on the system parameters (10). When (10) is violated, the same approach can be applied to individual pieces of cold videos (chunks), instead of entire videos, with less stringent constraint. Indeed, peers who are forced to contribute to an unrequested video, neither need to completely retrieve it, nor to download it sequentially. Thus, we can cut a cold video in M chunks, and ask some peers to download just a randomly chosen chunk contributing to its distribution. Chunkization reduces the bandwidth that every artificial downloader consumes by a factor M, while keeping constant its potential contribution on the upload.

E. Catalog warming by stripe bundling

A similar idea can be applied to stripes, instead of chunks, and superposed to the load balancing technique proposed in V-B. Essentially, peers who request for the first time a hot content $k \leq K_0$ (for some constant K_0), with probability $p'_k = \frac{\lambda'_k - \lambda_k}{\sum_{h \leq K_0} \lambda_h}$ are forced to download also a randomly chosen stripe of cold video k, contributing to its distribution for the rest of their stay into the system, with an opportunely chosen fraction of their upload bandwidth. Observe that by construction the load on every cold content can be maintained constant by properly selecting the fraction of peer upload bandwidth contributed to its distribution, while the load increase for hot videos (due to the subtraction of some upload bandwidth) vanishes as the system size increases, since $p'_d = \sum_{k>K_0} p'_k \to 0$ as $n \to \infty$. As a consequence this scheme can always be applied in sufficiently large systems.

F. Asymptotic bandwidth requested from the servers

At last, we can evaluate the asymptotic performance achievable by applying the active schemes described in previous sections.

Theorem 5: Under the same assumptions on U of Theorem 1, if there exists an arbitrarily small constant $\sigma > 0$ such that $\mathbb{P}(\Gamma < 1 - \sigma) \to 1$ for some $\sigma > 0$, the bandwidth $\overline{S}$ requested from the servers satisfies the following asymptotic bound w.h.p. as the number of users n tends to ∞:

$$\overline{S} = \begin{cases} o(1) & \text{if } \alpha > 1, \beta < 1 \\ O(n^{2-\alpha} (\log n)^{\frac{\alpha-1}{1-\delta}}) & \text{if } \alpha > 1, \beta = 1 \end{cases} \tag{11}$$

provided that a suitable combination of active techniques is employed i) to balance the loads induce by individual videos; ii) to sufficiently increase the download rate of cold videos.

Due to the lack of space, the proof of Theorem 5 is reported in our technical report [13], which includes also a precise definition of the desired rates λ'_k and the constants needed to implement the catalog warming techniques. The proof of Theorem 5 goes essentially along the same lines as in the proof of Theorem 2.

VI. Lower Bound

Here we present a simple universal lower bound to the bandwidth requested from the servers. Notice however that this bound holds under the assumption that the size of chunk/stripe cannot go to zero. Consider first the case in which videos are not divided into chunk/stripes. For any video k the servers must provide at least a bandwidth equal to d_k when the following two conditions jointly occur: i) there is at least one user downloading the video; ii) there are no seeds assisting its distribution. Thus, we can write:

$$\overline{S} \quad \geq \quad \sum_{k=1}^{K} d_k \mathbb{P}(N_{d,k} > 0) \mathbb{P}(N_{\text{seed},k} = 0) \tag{12}$$

Previous argument can be extended to the case in which videos are divided into a finite number of chunks/stripes, considering every chunk/stripe as an individual object. By algebraically manipulating (12), we obtain:

Theorem 6: The average bandwidth requested from the servers, $\overline{S}$, satisfies the following asymptotic bound as the number of active users n tends to ∞:

$$\overline{S} = \begin{cases} \Omega\left(n^{2-\alpha}\right) & \text{if } \alpha > 1, \beta = 1 \\ \Theta(n) & \text{if } \alpha \leq 1, \beta = 1 \\ 0 & \text{if } \beta < 1 \end{cases} \tag{13}$$

We report the proof of Theorem 6 in our technical report [13]. Essentially the proof consists in finding a lower bound for (12), that uniformly holds under any possible distribution of seeds to videos (*i.e.*, satisfying $\sum_k \overline{N}_{\text{seed},k} = O(N)$).

VII. Conclusions

Our results indicate that users' cooperation can dramatically reduce the servers' burden in large-scale VoD systems. Although peer-assisted architectures incur several issues related to the added complexity on the control plane, the need to provide incentive mechanism to the users and to protect the system against attacks and misbehavior, nevertheless we believe they should be taken seriously into consideration in the coming years, as they are the only known solution (up to now) to make VoD systems arbitrarily scalable. However, we have shown that the potential gains deriving by users' cooperation are reduced when the service is targeted to the distribution of user-generated contents (especially for small values of the Zipf's law exponent), since in this case the number of videos intrinsically scales linearly with the number of users.

References

[1] "Cisco Visual Networking index: Forecast and Methodology, 2011–2016," White paper published on Cisco web site, 2012.

[2] D. Niu, H. Xu, B. Li, and S. Zhao, "Quality-Assured Cloud Bandwidth Auto-Scaling for Video-on-Demand Applications," in *INFOCOM*, 2012.

[3] C. Huang, J. Li, and K. Ross, "Can Internet Video-on-Demand Be Profitable?" in *ACM SIGCOMM*, 2007.

[4] Y. Huang, T. Z. J. Fu, D. ming Chiu, J. C. S. Lui, and C. Huang, "Challenges, Design and Analysis of a Large-scale P2P VoD System," in *ACM SIGCOMM*, 2008.

[5] PPLive, http://www.gridcast.cn/. GridCast,http://www.gridcast.cn/. PPStream,http://www.ppstream.com/. TVU,http://www.tvunetworks.com/.

[6] W. Wu, R. Ma, and J. Lui, "On Incentivizing Caching for P2P-VoD Systems," in *NetEcon Workshop*, 2012.

[7] F. Dobrian, V. Sekar, A. Awan, I. Stoica, D. Joseph, A. Ganjam, J. Zhan, and H. Zhang, "Understanding the impact of video quality on user engagement," in *ACM SIGCOMM*, 2011.

[8] D. Ciullo, V. Martina, M. Garetto, E. Leonardi, and G. L. Torrisi, "Stochastic Analysis of Self-Sustainability in Peer-Assisted VoD Systems," in *IEEE INFOCOM*, 2012.

[9] D. Wu, Y. Liu, and K. Ross, "Modeling and Analysis of Multichannel P2P Live Video Systems," *IEEE/ACM Trans. Netw.*, vol. 18(4), 2010.

[10] W. Wu and J. C. Lui, "Exploring the optimal replication strategy in p2p-vod systems: Characterization and evaluation," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, pp. 1492–1503, 2012.

[11] P. Dhungel, K. Ross, M. Steiner, Y. Tian, and X. Hei, "Xunlei: Peer-Assisted Download Acceleration on a Massive Scale," in *PAM*, 2012.

[12] M. Cha, H. Kwak, P. Rodriguez, Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: Analyzing the world's largest user generated content video system," in *IMC*, 2007.

[13] Companion technical report, available at http://www.telematica.polito.it/~emilio/papers/Infocom13.pdf.

Appendix A
Proof of Theorem 1

The proof of this theorem exploits some intermediate results from [8], which are briefly recalled. The following upper bound holds for the bandwidth requested from the servers:

$$\overline{S} \leq \left(d + \frac{1}{\theta^*} e^{-\theta^* A} e^{\overline{N}_d (e^{\theta^* A} - 1)}\right) e^{-\overline{N}_{\text{seed}}(1-\phi_U(-\theta^*))} e^{\theta^* d} \quad (14)$$

where ϕ_U be the moment generating function of U, ϵ is an arbitrary positive constant less than $\overline{U}$ and $A \triangleq d - \overline{U} + \epsilon$. Moreover, θ^* is the unique strictly positive solution to the equation $\mathbb{E}[e^{\theta(d-U-A)}] = \mathbb{E}[e^{\theta(\overline{U}-U-\epsilon)}] = 1$.

The following two propositions from [8] establish important properties of θ^* as function[6] of ϵ:

Proposition 1: If $d > \overline{U}$, the equation $\mathbb{E}[e^{\theta(\overline{U}-U-\epsilon)}] = 1$ (in θ) admits a unique solution for $\epsilon \in (0, \overline{U})$. Furthermore, $\theta^*(\epsilon) = \arg_{\theta>0}(e^{-\theta\epsilon}\mathbb{E}[e^{\theta(\overline{U}-U)}] = 1)$ is strictly increasing and C^1 on the interval $(0, \overline{U})$. Moreover, it holds $\lim_{\epsilon\to 0} \theta^*(\epsilon) = 0$.

[6] In the following, whenever not necessary, we use θ^* instead of $\theta^*(\epsilon)$.

Proposition 2: Provided that $d > \overline{U}$, and U is not constant, the image of $\theta^*(\epsilon)$ for $0 < \epsilon < \overline{U}$ is $\mathbb{R}^+ \setminus \{0\}$.

Our goal is to tightly characterize the asymptotic behavior of bound (14) as $\lambda \to \infty$. We first focus on the case $d > \overline{U}$. In this case we will make $\epsilon \to 0$ as $\lambda \to \infty$, exploiting Propositions 1 and 2.

Consider, first, quantity $e^{-\theta^* A} e^{\overline{N}_d (e^{\theta^* A} - 1)}$ in (14). Note that $e^{-\theta^* A} \leq 1$, and $e^{\theta^* A} - 1 > \theta^* A$. Thus, for all $\eta \in (0, 1)$, there exists $\theta_0^*(\eta) > 0$ such that if $\theta^* \in (0, \theta_0^*)$ then $e^{\theta^* A} - 1 \leq \theta^* A/(1-\eta)$. In particular, $\theta_0^*(\eta) = -\frac{1-\eta+\mathbb{W}((\eta-1)e^{\eta-1})}{A}$, where $\mathbb{W}(\cdot)$ is the Lambert function.We obtain:

$$e^{-\theta^* A} e^{\overline{N}_d (e^{\theta^* A} - 1)} \leq e^{\overline{N}_d \frac{\theta^* A}{1-\eta}}. \quad (15)$$

Consider now quantity $e^{-\overline{N}_{\text{seed}}(1-\phi_U(-\theta^*))}$ in (14). Since for $\epsilon \to 0$ we have $\theta^* \to 0$ (Proposition 1), $1 - \phi_U(-\theta^*) = 1 - \mathbb{E}[e^{-\theta^* U}] = \theta^* \overline{U} + R(\theta^*)$, where $R(\theta^*)$ is the Taylor remainder. If we express $R(\theta^*)$ in the Lagrange form we immediately obtain: $|R(\theta^*)| \leq \mathbb{E}[U^2]\theta^{*2}/2$.

After some elementary algebra it is possible to show that for every $\eta' > 0$, defining $\theta_1^*(\eta') = \frac{\eta'}{1+\eta'} \frac{2\overline{U}}{\mathbb{E}[U^2]}$ we have that if $\theta^* \in (0, \theta_1^*)$ it holds: $1 - \eta' \leq \dfrac{\theta^* \overline{U}}{1 - \phi_U(-\theta^*)} \leq 1 + \eta'$, and therefore

$$1 - \phi_U(-\theta^*) \geq \frac{\theta^* \overline{U}}{1 + \eta'}. \quad (16)$$

Consider now quantity $e^{\theta^* d}$ in (14): defining $\theta_2^* = (\log 2)/d$, it is immediate to see that for $\theta^* \in (0, \theta_2^*)$, it holds:

$$e^{\theta^* d} \leq 2. \quad (17)$$

By (14), (15), (16) and (17), we can conclude that for all $\theta^* \in (0, \min\{\theta_0^*, \theta_1^*, \theta_2^*\})$, it holds

$$\overline{S} \leq 2 \left(d + \frac{e^{\overline{N}_d \frac{\theta^* A}{1-\eta}}}{\theta^*}\right) e^{-\overline{N}_{\text{seed}} \frac{\theta^* \overline{U}}{1+\eta'}} \quad (18)$$

From (3) we can derive a relation between the number of downloaders, $\overline{N}_d = \lambda\tau$, and the number of seeds:

$$\overline{N}_{\text{seed}} = \left(\frac{d}{\overline{U}\gamma} - 1\right) \overline{N}_d = \left(\frac{d}{\overline{U}\gamma} - 1\right) \lambda\tau \quad (19)$$

Substituting (19) in (18) we get:

$$\overline{S} \leq 2 \left(d + \frac{e^{\lambda\tau \frac{\theta^* A}{1-\eta}}}{\theta^*}\right) e^{-\left(\frac{d}{\overline{U}\gamma} - 1\right)\lambda\tau \frac{\theta^* \overline{U}}{1+\eta'}} \quad (20)$$

Recall from Proposition 1 that, for $d > \overline{U}$ and $\epsilon \to 0$, we have $\theta^* \to 0$. Moreover, by Propositions 1 and 2, as $\lambda \to \infty$, we can set $\theta^* \sim \lambda^{-\delta}$ (*i.e.*, we can find the proper law for ϵ that leads to $\theta^* \sim \lambda^{-\delta}$) for all $\delta \in (0, 1)$.

Thus, there exists a $\lambda_0 > 0$ such that $\forall \lambda > \lambda_0$, we have that $\theta^* \in (0, \min\{\theta_0^*, \theta_1^*, \theta_2^*\})$. Then, if $\lambda > \lambda_0$, for all $\eta, \eta' > 0$ we have:

$$\begin{aligned} \overline{S} &\leq 2 \left(d + \lambda^\delta e^{\lambda^{1-\delta} \tau \frac{A}{1-\eta}}\right) e^{-\left(\frac{d}{\overline{U}\gamma} - 1\right)\tau \frac{\overline{U}}{1+\eta'} \lambda^{1-\delta}} = \\ &= 2\, d\, e^{-\lambda^{1-\delta}\tau \frac{d-\overline{U}\gamma}{\gamma(1+\eta')}} + 2\lambda^\delta e^{-\lambda^{1-\delta}\tau \frac{(d-\overline{U}\gamma)(1-\eta) - A\gamma(1+\eta')}{(1+\eta')(1-\eta)\gamma}} \end{aligned} \quad (21)$$

Remembering that $A \triangleq d - \overline{U} + \epsilon$, for all $\epsilon > 0$, $\eta, \eta' \in (0, 1)$ we define the quantity

$$f(\epsilon, \eta, \eta', d, \gamma, \overline{U}) \triangleq \tau \frac{(d - \overline{U}\gamma)(1-\eta) - A\gamma(1+\eta')}{(1+\eta')(1-\eta)\gamma} \quad (22)$$

that appears in (21). We compute now $f(\epsilon,\eta,\eta',d,\gamma,\overline{U})$ for $\epsilon,\eta,\eta'\to 0$:

$$\lim_{\epsilon,\eta,\eta'\to 0} f(\epsilon,\eta,\eta',d,\gamma,\overline{U}) = \tau\frac{d(1-\gamma)}{\gamma} > 0.$$

Since $f(\epsilon,\eta,\eta',d,\gamma,\overline{U})$ for $\gamma\neq 0$ and $\eta\neq 1$ is a continuous function and hence a uniformly continuous one over compact sets that do not contain points either with $\gamma=0$ or $\eta=1$, three strictly positive constants $\epsilon_0,\eta_1,\eta_2>0$ can be found such that, for all $0<\epsilon<\epsilon_0$, $0<\eta<\eta_1$ and $0<\eta'<\eta_2$, we have

$$f(\epsilon,\eta,\eta',d,\gamma,\overline{U}) > \frac{1}{2}\tau\frac{d(1-\gamma)}{\gamma} > 0.$$

uniformly with respect to parameters d, γ and $\overline{U}$ as long as they take values over a compact set that does not contain the points with $\gamma=0$ (as in our case). Defining the constant

$$C_1 \triangleq \frac{1}{2}\tau\frac{d(1-\gamma)}{\gamma}, \tag{23}$$

we can conclude that, uniformly with respect to the parameters, for $d>\overline{U}$ it holds:

$$\overline{S} = O\left(\lambda^{\delta}e^{-C_1\lambda^{1-\delta}}\right) \quad \text{as } \lambda\to\infty \tag{24}$$

since the fist term in (21) is negligible with respect to the second one, in light of the fact that $\tau\frac{d-\overline{U}\gamma}{\gamma(1+\eta')} > C_1 = \frac{1}{2}\tau\frac{d(1-\gamma)}{\gamma}$.

Consider now the case $d<\overline{U}$. From Theorem 1 in paper [8], we have that θ^* is a constant as $\lambda\to\infty$, and we can set ϵ in such a way that $A=0$. Using a similar reasoning as for the case $d>\overline{U}$, after some calculations we get

$$\overline{S} \le C_3 e^{-C_2\lambda} \qquad \text{as } \lambda\to\infty, \tag{25}$$

where $C_3 \triangleq (d+1/\theta^*)e^{\theta^* d}$ and
$C_2 \triangleq \left(\frac{d}{\overline{U}\gamma}-1\right)\tau(1-\phi_U(-\theta^*))) = \left(\frac{d}{\overline{U}\gamma}-1\right)\tau(1-e^{-\theta^* d})$.

At last we consider the case $\gamma>1$ (*i.e.*, the deficit mode). In this case the bandwidth requested from servers scales as $\Theta(n)$, as shown in [8]. Indeed, the servers have to provide at least the bandwidth deficit.

Appendix B
Proof of Theorem 2

First we assume $\mathbb{P}(\Gamma_k < 1-\sigma) = 1$. In this case all γ_k are deterministically smaller than $1-\sigma$, and by Corollary 2 for each video we get:

$$\overline{S}_k \le \begin{cases} d_k\tau_k\lambda_k & \text{if } \lambda_k<\lambda_0 \\ 2\lambda_k^{\delta}e^{-C_{1,k}\lambda_k^{1-\delta}} \ \forall\delta\in(0,1) & \text{if } \lambda_k\ge\lambda_0, \end{cases} \tag{26}$$

where $C_{1,k} \triangleq C_1(d_k,\tau_k,\gamma_k')$, $\gamma_k' = (1+\sigma/2)\gamma_k$. We divide videos in two categories, depending on the request rate λ_k. Video k belongs to the first category if $0\le k\le K_1$, with K_1 such that $\lambda_{K_1}=\lambda_0$, where λ_0 is the threshold defined in Corollary 2. We thus obtain that $K_1 = \lambda_0^{-1/\alpha}n^{1/\alpha}H(K)^{1/\alpha}$. We distinguish the following cases depending on α:

$$K_1 = \begin{cases} \Theta(n^{1/\alpha}) & \text{if } \alpha>1 \\ \Theta(n\log n) & \text{if } \alpha=1 \\ \Theta(n^{\frac{1+\beta(\alpha-1)}{\alpha}}) & \text{if } \alpha<1 \end{cases} \tag{27}$$

Videos k such that $K_1\le k\le K$ belong to the second category. Comparing asymptotically K_1 with K, we obtain:

$$\begin{array}{ll} K_1 = o(K) & \text{if } \alpha\ge 1, 1/\alpha<\beta\le 1 \\ K_1 = \omega(K) & \text{if } \alpha\ge 1,\beta\le 1/\alpha \text{ or } \alpha<1,\beta\le 1 \end{array} \tag{28}$$

Therefore, when $\alpha<1$, we set $K_1\equiv K$, and we have only one video category. Now, to compute $\overline{S}$ we can just sum up the contributions of all videos, obtaining:

$$\begin{array}{rcl} \overline{S} & = & \sum_{k=1}^{K}\overline{S}_k \\ & \le & \sum_{k=1}^{K_1-1}2\lambda_k^{\delta}e^{-C_{1,k}\lambda_k^{1-\delta}} + \sum_{k=K_1}^{K} d_k\tau_k\lambda_k. \end{array} \tag{29}$$

We define $C \triangleq d_{\max}\tau_{\max}$, and substitute λ_k with its value $np_k = nH(K)k^{-\alpha}$. We obtain:

$$\begin{array}{rcl} \overline{S} & \le & 2(nH(K))^{\delta}\sum_{k=1}^{K_1-1}k^{-\alpha\delta}e^{-C_{1,k}(nH(K)k^{-\alpha})^{1-\delta}} \\ & & + C\,n\,H(K)\sum_{k=K_1}^{K}k^{-\alpha}. \end{array} \tag{30}$$

Let $\overline{S}_{up,1}$ be the first term in (30). Furthermore, since by (23) $C_{1,k} \triangleq \frac{1}{2}\tau_k\frac{d_k(1-\gamma_k)}{\gamma_k}$, we have $C_{1,\inf} \triangleq \inf_k C_{1,k} \ge \frac{1}{2}\sigma\tau_{\min}d_{\min}$. Thus, we have:

$$\begin{array}{l} \overline{S}_{up,1} \le 2\sum_{k=1}^{K_1-1}(nH(K)k^{-\alpha})^{\delta}e^{-C_{1,\inf}(nH(K)k^{-\alpha})^{1-\delta}} \\ < \Theta\left(\int_1^{K_1-1}(nH(K)x^{-\alpha})^{\delta}e^{-C_{1,\inf}(nH(K)x^{-\alpha})^{1-\delta}}dx\right) \end{array}$$

Now we make the substitution $y = (nH(K)x^{-\alpha})^{1-\delta}$ and get $dx = \frac{(nH(K))^{\frac{1}{\alpha}}}{\alpha(1-\delta)y^{1+\frac{1}{\alpha(1-\delta)}}}dy$. We have:

$$\overline{S}_{up,1} < \Theta\left(\frac{(nH(K))^{\frac{1}{\alpha}}}{\alpha(1-\delta)}\int_{(nH(K)(K_1(n)-1)^{-\alpha})^{1-\delta}}^{(nH(K))^{1-\delta}} e^{-yC_{1,\inf}}\frac{y^{\frac{\delta}{1-\delta}}}{y^{1+\frac{1}{\alpha(1-\delta)}}}dy\right)$$

If $y^{\frac{2\alpha\delta-\alpha-1}{\alpha(1-\delta)}} < 1$, that is if $\delta < 1/(2\alpha)+1/2$, we obtain

$$\begin{array}{l} \overline{S}_{up,1} < \Theta\left(\frac{(nH(K))^{\frac{1}{\alpha}}}{\alpha(1-\delta)}\int_{(nH(K)(K_1-1)^{-\alpha})^{1-\delta}}^{(nH(K))^{1-\delta}} e^{-yC_{1,\inf}}dy\right) \\ < \Theta\left((nH(K))^{\frac{1}{\alpha}}e^{-C_{1,\inf}(nH(K)(K_1(n))^{-\alpha})^{1-\delta}}\right) \end{array}$$

From (27) and (28), we get:

$$\overline{S}_{up,1} < \begin{cases} \Theta(n^{\frac{1}{\alpha}}e^{-C_{1,\inf}}) = O(n^{1/\alpha}) & \text{if } \alpha>1,\beta>1/\alpha \\ \Theta(n^{\frac{1}{\alpha}}e^{-C_{1,\inf}n^{(1-\alpha\beta)(1-\delta)}}) = o(1) & \text{if } \alpha>1,\beta<1/\alpha \\ \Theta(n\log n e^{-C_{1,\inf}}) = O(n\log n) & \text{if } \alpha=1,\beta=1 \\ \Theta(n\log n^{\beta}e^{-C_{1,\inf}(n^{1-\beta}\log n^{\beta})^{(1-\delta)}}) = o(1) & \text{if } \alpha=1,\beta<1 \\ \Theta(ne^{-C_{1,\inf}}) = O(n) & \text{if } \alpha<1,\beta=1 \\ \Theta(n^{\beta(\alpha-1)+1}e^{-C_{1,\inf}n^{1-\beta}}) = o(1) & \text{if } \alpha<1,\beta<1 \end{cases}$$

Now we consider the second term in (30), $\overline{S}_{up,2} \triangleq CnH(K)\sum_{k=K_1}^{K}k^{-\alpha}$. Note that this term exists only when $K_1 = o(K)$, see (28). Since function $f(x)=x^{-\alpha}$ is decreasing, by the integral test for series we obtain, for $\alpha>1$,

$$\overline{S}_{up,2} \;<\; nH(K)\left(K_1^{-\alpha}+\int_{K_1}^{K}x^{-\alpha}\,\mathrm{d}x\right) = \Theta\left(n^{\frac{1}{\alpha}}\right)$$

If $\alpha=1$, with the same calculation as above we obtain

$$\overline{S}_{up,2} \;<\; \Theta\left(n\log^2 n\right)$$

Noting that a trivial upper bound to $\overline{S}$ is n, given the above bounds on $\overline{S}_{up,1}$ and $\overline{S}_{up,2}$, we obtain:

$$\overline{S} = \begin{cases} O(n^{1/\alpha}) & \text{if } \alpha>1,\beta>1/\alpha \\ o(1) & \text{if } (\alpha>1,\beta<1/\alpha)\vee(\alpha\le 1,\beta<1) \\ O(n) & \text{if } \alpha\le 1,\beta=1 \end{cases}$$

When, instead, $\mathbb{P}(\Gamma_k\le 1)<1$, for any $\beta>0$ standard concentration arguments allow to say that for any function $f(n)\to\infty$, a finite fraction of videos with index $k\le f(n)$ will have w.h.p. an associated load $\gamma_k>1$. Since the associated request rate for such videos scales linearly with n (*i.e.*, it scales faster that any sub-linear function), as immediate consequence $\overline{S}$ scales also linearly with n.