



**HAL**  
open science

## A comprehensive analytical framework for VoD services in hybrid CDN-P2P systems

Noé Torres-Cruz, Mario Rivero-Angeles, Gerardo Rubino, Ricardo  
Menchaca-Mendez, Rolando Menchaca-Mendez, David Ramirez

► **To cite this version:**

Noé Torres-Cruz, Mario Rivero-Angeles, Gerardo Rubino, Ricardo Menchaca-Mendez, Rolando Menchaca-Mendez, et al.. A comprehensive analytical framework for VoD services in hybrid CDN-P2P systems. Journal of Network and Computer Applications (JNCA), 2020, 161, pp.1-17. 10.1016/j.jnca.2020.102643 . hal-03122777

**HAL Id: hal-03122777**

**<https://hal.inria.fr/hal-03122777>**

Submitted on 5 Jul 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A comprehensive analytical framework for VoD services in hybrid CDN-P2P systems

Noé Torres-Cruz <sup>a,\*</sup>, Mario E. Rivero-Angeles <sup>a</sup>, Gerardo Rubino <sup>b</sup>,  
Ricardo Menchaca-Mendez <sup>a</sup>, Rolando Menchaca-Mendez <sup>a</sup>, David Ramirez <sup>c</sup>

<sup>a</sup> CIC, Instituto Politécnico Nacional, Av. Juan de Dios Bátiz, Esq. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C. P. 07738, Ciudad de México, Mexico

<sup>b</sup> INRIA Rennes - Bretagne Atlantique, Campus Universitaire de Beaulieu, 35042, Rennes Cedex, France

<sup>c</sup> Department of Electrical Engineering, Princeton University, USA

## ABSTRACT

Video on Demand services generate the largest amount of traffic in networks nowadays. Previous works have proposed integrating Content Delivery Networks (CDN) and Peer-to-Peer (P2P) networks to satisfy this demand. However, their analytical methods do not consider all the factors that affect the performance of these systems. Hence, we present a novel comprehensive framework, based on a fluid model, to evaluate VoD services over hybrid CDN-P2P systems. The proposed framework considers features of the service (e.g., size, coding-rate and popularity of the video), network attributes (e.g., upload data-rate capacity of servers and peers) and characteristics of the behavior of the users (e.g., sojourn time, cooperativeness and frequency of random-seeks). Our framework allows a system to be evaluated under a wide variety of scenarios in terms of network-cost and Quality-of-Service (QoS) parameters and is flexible enough to model different resource allocation schemes, including a novel scheme to distribute the upload network capacity. Despite the wide variety of considered factors, our framework is tractable and as accurate as discrete-models based on Markov chains.

Currently, video services account for an unprecedented quantity and portion of Internet data traffic. In 2018, video services generated 56% of the global data traffic primarily from Video on Demand (VoD) services such as Netflix, YouTube and Amazon Prime ([The Global Internet Phenomena Report, 2019](#)). Moreover, video services are expected to occupy an estimated 74% of mobile data traffic by 2024 ([Ericsson Mobility Report, 2018](#)). Specifically, VoD services have boomed because they allow subscribers ubiquitous access to immense content libraries, and to initiate a video playback even if this video has not been fully downloaded.

Content of VoD systems is stored before distribution, hence, they can take advantage of caching strategies implemented through Content Delivery Networks (CDNs) and Peer-to-Peer (P2P) networks. In CDNs, servers distributed over multiple Internet Service Providers (ISPs) push

content close to end users to reduce delay and improve the Quality-of-Service (QoS); unfortunately, with the ever-increasing amount of video traffic, CDNs still face significant scalability problems ([Zhang et al., 2015](#)). On the other hand, P2P networks have shown to scale much better, but their performance can be degraded by phenomena associated with users behaviors such as free-riding (non-cooperative peers) and churning (departures of serving peers). In this context, hybrid CDN-P2P systems have proven to be very effective because they exhibit the scalability of the P2P networks and the stability provided by the fixed infrastructure of the CDNs ([Zhang et al., 2015](#); [Li et al., 2015](#)).

Modeling VoD services over hybrid networks is a challenging task because the performance of such systems depends on a wide variety of factors. These factors include characteristics of video-files (e.g., popularity, coding rate and size), properties of both CDNs (e.g., upload capacity) and peers (e.g., upload and download data rate, sojourn time), as well as the implemented schemes to allocate the available resources.

\* Corresponding author.

E-mail addresses: [ntorresc@ipn.mx](mailto:ntorresc@ipn.mx) (N. Torres-Cruz), [mriveroa@ipn.mx](mailto:mriveroa@ipn.mx) (M.E. Rivero-Angeles), [gerardo.rubino@inria.fr](mailto:gerardo.rubino@inria.fr) (G. Rubino), [ric@cic.ipn.mx](mailto:ric@cic.ipn.mx) (R. Menchaca-Mendez), [rmenc@cic.ipn.mx](mailto:rmenc@cic.ipn.mx) (R. Menchaca-Mendez), [dard@princeton.edu](mailto:dard@princeton.edu) (D. Ramirez).

<https://doi.org/10.1016/j.jnca.2020.102643>

Many works have analyzed VoD systems but, in general, have not comprehensively modeled all the above enlisted factors (Ciullo et al., 2014; Romero et al., 2015; Gramatikov and Jaureguizar, 2016; Hwang et al., 2016; Sasabe, 2018; Baez-Esquivel et al., 2013; Torres-Cruz et al., 2017; Torres-Cruz et al., 2018; Tian et al., 2015; Cong and Shuang, 2015; Zhao et al., 2015; Saengarunwong and Sanguankotchakorn, 2018). Some of these works propose schemes to improve the system performance, but do not develop analytical tools (Cong and Shuang, 2015; Zhao et al., 2015). Some others do propose analytical models, but are restricted to pure P2P networks (Hwang et al., 2016; Baez-Esquivel et al., 2013; Torres-Cruz et al., 2018) or, despite analyzing hybrid networks, do not consider all the elements involved in this kind of systems (Ciullo et al., 2014; Romero et al., 2015; Gramatikov and Jaureguizar, 2016; Torres-Cruz et al., 2017). In Section 2, we provide a more detailed description of the restrictions of these works.

On the basis of the previous arguments, we propose a comprehensive analytical framework that is based on a fluid model. Our model considers the population dynamics of a set of peers sharing a particular video. The video-file is divided into fragments, numbered according to their playing order, which we call *windows*. Peers are classified by the indexes of their current downloading ( $j$ ) and playing ( $k$ ) windows. The model describes the *flow* of peers among the different  $(j, k)$ -groups by considering all the factors listed in the previous paragraph.

Using our fluid model, we can estimate the number of peers in each  $(j, k)$ -group, which allows us to calculate the average download data rate demanded by each of these groups. Then, given a resource allocation scheme, we are able to obtain the amount of upload data rate that must be provided by the CDN or by the peers to supply each group demand. Notice that these data rates are directly related to both the operating cost of the CDN and the amount of resources that must come from the P2P network (e.g., energy and upload bandwidth (Deltouzos and Denazis, 2015)).

The proposed framework also allows us to evaluate the number of users experiencing startup delays (i.e., time elapsed between the video request and the beginning of the video playing) or stalling events (i.e., playback interruptions because the peer has played all its buffered content), which are relevant QoS parameters for VoD (Hossfeld et al., 2012). Therefore, our model is an effective tool to analyze trade-offs between network costs and QoS parameters.

Leveraging the flexibility of our model, we develop a variation that incorporates the process of pausing and resuming the downloading process in terms of the non-played buffer-content. Additionally, we present a second variation that considers VCR functions (e.g., pause, fast forward, or rewind). These model variations allow us to evaluate the costs and performance of the network under more realistic scenarios.

In addition, our model enables to evaluate resource allocation schemes (which have a major impact over the system performance) by defining the upload capacity allocated to supply a particular window from the P2P network, and from the CDN. To showcase the capability of our model to evaluate schemes, we propose a novel one, where resources from a particular  $(j, k)$ -group are assigned with high priority to peers requesting immediate lower windows. We call this scheme *Generalized Prioritized-Windows Allocation* (GPWA). Experimental results show that GPWA can efficiently distribute the resources from peers, thus reducing required server resources in comparison with a representative of the state-of-the-art (Zhao et al., 2016).

Since the fluid model describes the number of peers as a continuous variable, we complement our analysis with a Markov chain model, whose state is defined as the vector containing the number of peers in each  $(j, k)$  group. Numerical evaluations show consistency between our fluid-model at steady-state and the Markov chain model when upload data rate is equal or larger than demanded data rate (a mandatory condition to satisfy QoS targets).

The main contribution of our work is the development of a comprehensive and tractable framework that allows us to evaluate the per-

formance of VoD services over CDN-P2P networks in a wide range of practical scenarios and in terms of network-cost and QoS parameters. Moreover, this framework is flexible enough to study different resource allocation schemes, including a novel proposal to distribute the upload network capacity, which is more efficient than previous schemes.

The rest of this work is organized as follows. In Section 3, we introduce our fluid model for a basic scenario, where only continuous-and-ordered downloading and playback processes are considered. In Section 4, we describe our proposed resource allocation scheme, namely, the GPWA scheme. In Section 6, we present an extension of the fluid model to include restrictions over the downloading process in terms of the buffer state, and, in Section 7, we further incorporate the effects of VCR functions. In Section 8, we present the Markov chains associated to each model variation. In Section 9, we define a set of QoS parameters that can be directly obtained from the fluid model solution. Lastly, in Sections 10 and 11, we present our numerical results and conclusions, respectively.

## 2. Related work

There exists a large body of research devoted to analyze VoD services over P2P or CDN-P2P networks. In works like (Wu et al., 2014; Wichtlhuber et al., 2015), the authors propose strategies to incentivize cooperation among peers. On the other hand, there are multiple papers devoted to design *caching policies*, i.e., strategies to storage contents in order to make them as accessible as possible to end users, some examples of these proposals are (Faiqurahman and Kistijantoro, 2015; Wu et al., 2014; Zhou et al., 2015). Furthermore, several strategies to allocate resources have been recently proposed, this kind of works can be classified into two main categories: algorithms that define, for a specific user, from which peers obtain the required content in order to improve his or her QoS (Hwang et al., 2016; Sasabe, 2018; Zhao et al., 2016; Bethanabhotla et al., 2015; Dubin et al., 2015), and schemes that specify how to efficiently allocate the available bandwidth among peers or groups of peers in the system (Ciullo et al., 2014; Gramatikov and Jaureguizar, 2016; Torres-Cruz et al., 2017; Jia et al., 2016; Rohmer et al., 2015; Huang et al., 2014; Chang et al., 2013). In this paper, we propose a novel scheme that belongs to the latter of the previous classifications.

In addition, several evaluation frameworks have been recently developed. In the remaining of this section, we discuss the frameworks that are closely related to our work, i.e., those with a focus on fluid and Markov chain models.

De Veciana and Yang (2003) model a P2P network using a bi-dimensional Markov chain, where states are defined by the number of downloaders and seeders in the network. Following this approach, Qiu and Srikant (2004) developed a deterministic (*fluid*) model of a P2P network that provides simple expressions for the number of downloaders and seeders. Neither work considers video services, but have inspired recent work, e.g., (Ciullo et al., 2014; Romero et al., 2015; Gramatikov and Jaureguizar, 2016; Hwang et al., 2016; Sasabe, 2018; Baez-Esquivel et al., 2013; Torres-Cruz et al., 2017) discussed in the following.

Ciullo et al. (2014) propose a modeling framework to compute the required server bandwidth that prevents playback interruptions (i.e., stalling events) in a hybrid network. Their framework classifies peers either as downloaders or seeders and the authors consider scenarios such as non-stationary traffic, peer upload bandwidth heterogeneity, churning, and non-sequential chunk delivery. Despite the wide range of scenarios that this framework supports, in contrast to our proposal, it does not analyze how the strategies to allocate the upload bandwidth from peers affect the system performance.

Romero et al. (2015), evaluate a P2P network assisted by super-peers (i.e., fixed seeders) by means of fluid models. They consider that a peer can store multiple videos and their model describes the number of peers in possession of each one of such videos. The paper proves that the proposed model is globally stable in the Lyapunov sense. Since

this work was proposed for a system with multiple videos, it does not capture the details on how video fragments are shared among peers. The description of such a process is one of our main contributions.

Gramatikov and Jaureguizar (2016) also analyze a P2P network assisted by servers. The authors model the system through three interdependent Markov chains, whose states are defined in terms of the number of peers serving the video, the number of peers receiving it and the number of peers not participating in the process, respectively. After solving this network in stationary state, they evaluate the portion of streamed traffic from both servers and peers. Despite the relevance of this work, it makes some assumptions that restrict the kind of analyzed peers (e.g. it does not consider arrival and departure processes, since the analyzed peers are Set-Top Boxes), whereas in our proposal, we develop more general models.

Building on De Veciana and Yang, 2003, Hwang et al. (2016) develop a bidimensional Markov chain to evaluate a pure P2P network. The chain state is defined as the number of downloaders and peers in the system, and steady state solution is utilized to estimate the network download rate in terms of the seeders departure rate and the number of cached videos per peer. On the basis of these analyses, Hwang, et al. propose a protocol (i.e., Joint-Family) that implements chunk, bit-rate, and peer selection policies that minimize occurrence of playback interruptions. By contrast to our proposal, the model in Hwang et al. (2016) does not consider random seeks and does not specifically describe how video fragments are shared among peers.

Sasabe (2018) presents a discrete model which describes the *piece flow* among peers and servers in a Tit-for-tat-based P2P streaming system. Similar to other proposals, the model specifies how leeches turn into seeders and how these seeders abandon the system. Through integer linear programming, the model is solved in order to minimize the average play-out delay (startup delays and stalling events), the maximum play-out delay, and the average file retrieving time. It is important to notice that in Sasabe (2018) the system is analyzed on a peer-basis, whereas our framework describe it in terms of peers' populations. We consider that the latter approximation is more practical when analyzing systems with a huge number of peers (which is the case for very popular videos).

In Baez-Esquivel et al. (2013), Baez-Esquivel et al. analyze a pure P2P network, where video files are divided into fragments (called *windows*) and peers are grouped exclusively by the window they are currently downloading. From this classification, Baez-Esquivel et al. develop a fluid model as well as an associated Markov chain whose solutions provide the number of peers in each group; however, besides these results, no other performance parameters are presented, and the model is restricted to very specific scenarios. In this article, we overcome such limitations.

In Torres-Cruz et al. (2017), Torres et al. also develop a fluid model for a window-based VoD service, but, in contrast with (Baez-Esquivel et al., 2013), Torres et al. consider that the P2P network is assisted by servers and that the size of the first window is variable. The latter of these modifications introduces a trade-off between initial delay (i.e., startup delay) and the duration of playback interruptions (i.e., stalling events). In addition to the fluid model, Torres et al. use probabilistic analysis to determine the values of a set of operation parameters that guarantee the satisfaction of Quality-of-Experience (QoE) targets in the system. It should be notice that, different from this proposal, in Torres-Cruz et al. (2017) the fluid model does not consider the playback process, which restricts the scenarios and performance parameters that can be evaluated.

Inspired by previous work, we leverage fluid models and Markov chains to describe the evolution of the number of peers. In contrast with previous work, our proposed framework offers a more granular classification of peers, a wider range of service features, and a more detailed description of network dynamics. Thus, our analytical frame-

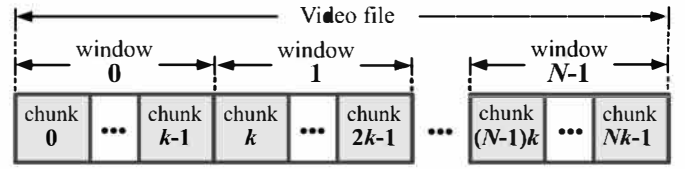


Fig. 1. Video-file divided in  $N$  windows, each containing  $k$  chunks.

work offers a holistic view of the interactions among peers and elucidates the impact of resource allocation strategies on network-costs and QoS in a wide variety of scenarios.

### 3. Basic two-dimensional scenario

#### 3.1. Model description

In this section, we describe the dynamics of a set of users that request the same video-file. We assume that the whole video is stored in content-servers that belong to the CDN infrastructure. Additionally, we consider that all the users can act as peers, i.e., they are capable of storing and uploading the video fragments that they have previously downloaded for their own consumption.<sup>1</sup> We further assume the existence of *tracker servers*, whose function is to establish connections between two peers, or between a peer and a content-server, according to a specific resource allocation scheme.

The shared video-file is divided into  $N$  equal-size fragments called *windows*. Windows are numbered according to their playing order and used as basic elements to model the evolution of the downloading and playing processes. Since video-files in VoD services are encoded in small fragments known as *chunks*, we propose that windows are composed of an integer number of chunks (see Fig. 1).

In this basic scenario, users can prefetch any number of windows. Due to the nature of videos, it is assumed that windows are downloaded and played in order. Peers are classified according to the index of the windows they are currently downloading and playing. Formally, group  $(j, k)$  is composed of all the peers that are currently downloading window  $j$  while playing window  $k$  where  $j, k \in [0, N]$  and  $k \leq j$ . Peers belonging to groups where  $j < N$  are known as *downloaders* or *leechers*. A peer downloading window  $j$  and suffering a stalling event (i.e., a playback interruption because the peer has played all its buffered windows) is classified in group  $(j, j)$ , for  $j \in [0 : N - 1]$ . We refer to these peers as *stalled leechers*, and, particularly, those in group  $(0, 0)$  are experiencing startup delay, which is unavoidable since *some* buffering is required before playback.

In P2P networks, peers that have downloaded the whole file and stay in the system are known as *seeders* because they can supply the file in its entirety to other peers. Particularly, in VoD services, a seeder may stay in the system because it has not finished playing the video. A seeder that is playing window  $k$  belongs to group  $(N, k)$ , hence,  $k < N$ ; while seeders that have finished playing video belong to group  $(N, N)$ . Note that seeders in the latter group have no inherent incentives to stay in the system and are considered *cooperative* seeders. Fig. 2 illustrates the peer classification in a Cartesian plane, where coordinates are associated with groups of peers.

The presented classification clearly distinguishes which windows are required for a peer and which windows it can share to other peers; thus eliminating the need to distribute buffer-maps to implement resource allocation schemes, as proposed in previous work (e.g., (Tian et al., 2015; Sheshjavani and Akbari, 2017)). In more complex scenarios, discussed in the following section, additional information is required to gain fine granular knowledge of the content required and possessed by each peer.

<sup>1</sup> In this paper, we use the terms “user” and “peer” interchangeably.



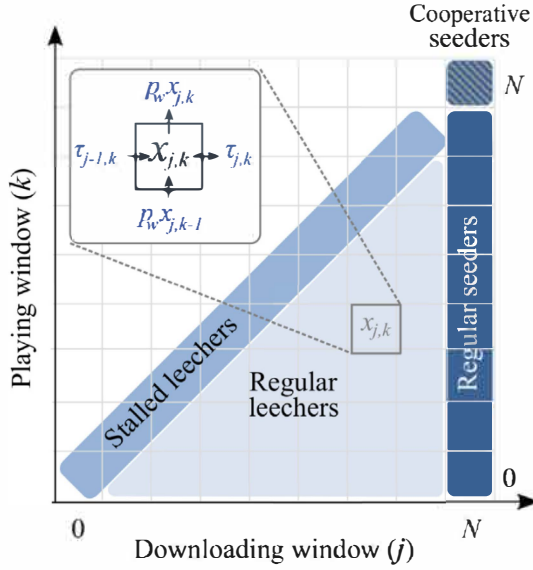


Fig. 2. Peer's classification in the basic scenario.

### 3.2. Fluid model for the basic scenario

Assume that peers arrive at the system with rate  $\lambda$ . The arrival rate depends on the popularity of the video. Additionally, assume that downloaders and regular seeders leave the system randomly at rate  $\theta$ , whereas cooperative seeders (which have finished the video playing) depart at rate  $\gamma \geq \theta$ . All these processes have, for an external observer, a random component, because of the uncountable factors that influence the arrival and departure processes to and from such a system. In our model, however, we will see them as physicists see a fluid, as deterministic quantities (see below).

For tractability, consider that video-files have a normalized size equal to 1. Denote the average download and upload data rates by  $c$  and  $\mu$  files/sec, respectively. These rates depend on the available upload/download bandwidth per peer and on the video coding rate (because they are given in file/sec), and we assume that all peers have the same characteristics in this regard.<sup>2</sup> Since the video is divided into  $N$  windows, the download and upload rates can also be given by  $c_w = Nc$  and  $\mu_w = N\mu$  in windows/sec, respectively. Assume that windows are played at rate  $p_w$  windows/sec. Last, define  $x_{j,k}(t)$  as the number of peers in group  $(j, k)$  at time  $t$ .

Considering these definitions, our model can account for the network dynamics, i.e., the rates at which peers are moving through groups or leaving the system. In general, as illustrated in Fig. 2, peers in group  $(j, k)$  experience the following transitions:

- As a result of the downloading process, peers transit to group  $(j + 1, k)$  at rate  $\tau_{j,k}$ . As explained below, the rate  $\tau_{j,k}$  depends on the available download rate for every group.
- Due to the playing process, peers transit to group  $(j, k + 1)$  at rate  $p_w x_{j,k}(t)$ .
- Peers leave the system at rate  $\theta x_{j,k}(t)$ .

In addition to these general transitions, the following special cases must be considered:

- Since we assume that users always play the video from the beginning, new peers enter the system exclusively through group  $(0, 0)$  at rate  $\lambda$ . This assumption is valid because after a request, VoD applications take the user to the beginning of the video. Even though skipping the beginning of the video is a very common practice, this phenomenon must be modeled as a random forward seek, as we explain in Section 7.1.
- Peers in group  $(j, j)$  (i.e., stalled leechers) cannot transit to group  $(j, j + 1)$  since they cannot play the video.
- Peers in group  $(N, k)$  cannot transit to group  $(N + 1, k)$  since they have completed the downloading process.
- Peers in group  $(N, N)$  cannot transit to another group because they have completed, both, the playing and downloading processes. Additionally, peers in group  $(N, N)$  leave the system at rate  $\gamma x_{N,N}(t)$ .

From the transitions detailed above, we establish the following set of differential equations in terms of the number of peers in each group:

$$x'_{0,0}(t) = \lambda - \theta x_{0,0}(t) - \tau_{0,0} \quad (1)$$

$$x'_{j,0}(t) = \tau_{j-1,0} - (\theta + p_w)x_{j,0}(t) - \tau_{j,0}; 0 < j < N \quad (2)$$

$$x'_{j,k}(t) = \tau_{j-1,k} + p_w x_{j,k-1}(t) - (\theta + p_w)x_{j,k}(t) - \tau_{j,k}; \quad (3)$$

$$0 < j < N, 0 < k < j$$

$$x'_{j,j}(t) = p_w x_{j,j-1}(t) - \theta x_{j,j}(t) - \tau_{j,j}; 0 < j < N \quad (4)$$

$$x'_{N,0}(t) = \tau_{N-1,0} - (\theta + p_w)x_{N,0}(t) \quad (5)$$

$$x'_{N,k}(t) = \tau_{N-1,k} + p_w x_{N,k-1}(t) - (\theta + p_w)x_{N,k}(t); \quad (6)$$

$$0 < k < N$$

$$x'_{N,N}(t) = p_w x_{N,N-1}(t) - \gamma x_{N,N}(t), \quad (7)$$

If all the peers in group  $(j, k)$  achieve the download rate  $c_w$ , then  $\tau_{j,k} = c_w x_{j,k}(t)$ , which means that the transition rate to group  $(j + 1, k)$  is limited only by the download demand, and we say that *abundance* conditions are satisfied for group  $(j, k)$ .

In contrast, if abundance conditions are not satisfied, then transition rate to group  $(j + 1, k)$  is limited by the allocated upload data rate for peers in group  $(j, k)$ , denoted by  $U(j, k)$ . In this case, we say that this group is under *penury* conditions, and  $\tau_{j,k} = U(j, k)$ . Since we focus on hybrid networks,  $U(j, k)$  includes upload capacity from both peers and content-servers and, as discussed in Section 3.3, also depends on the resource allocation scheme. Therefore,  $\tau_{j,k}$  can be written as

$$\tau_{j,k} = \min \{c_w x_{j,k}(t), U(j, k)\}, \quad (8)$$

It is important to emphasize that abundance conditions are necessary to guarantee appropriate levels of QoS (Torres-Cruz et al., 2018), because having smaller upload than download data rates inevitably leads to excessive download delays. Hence, we analyze the system under such conditions, and, in Section 3.3 we show how to remain within them given a specific resource allocation scheme.

Similar to (Qiu and Srikant, 2004; Ciullo et al., 2014; Romero et al., 2015; Baez-Esquivel et al., 2013; Torres-Cruz et al., 2017, 2018; Rivero-Angeles and Rubino, 2010), we focus on the steady-state of the model for two important reasons. First, per (Qiu and Srikant, 2004), when the request arrival rate  $\lambda$  is large, the fluid model in steady-state is a good approximation of the real system, e.g., when considering very popular videos. Second, Aspirot et al. (2011) demonstrates that the stochastic models (Markov chains) describing the system introduced in Rivero-Angeles and Rubino (2010) converge to deterministic

<sup>2</sup> In our model,  $c$  and  $\mu$  are constants because we assume that the video coding rate is the same along the downloading process; however, this can be easily modified by allowing the possibility of vary these values every window. This extension of the model will permit analyzing technologies like Dynamic Adaptive Streaming over HTTP (DASH).

limits (the steady-state of the fluid model) when the number of peers in the network is large.

By assuming abundance and steady-state conditions, from (1)-(7) we obtain the following system of linear equations:

$$\lambda - r_w X_{0,0} = 0 \quad (9)$$

$$c_w X_{j-1,0} - t_w X_{j,0} = 0; 0 < j < N \quad (10)$$

$$c_w X_{j-1,k} + p_w X_{j,k-1} - t_w X_{j,k} = 0; \quad (11)$$

$$0 < j < N, 0 < k < j$$

$$p_w X_{j,j-1} - r_w X_{j,j} = 0; 0 < j < N \quad (12)$$

$$c_w X_{N-1,0} - s_w X_{N,0} = 0 \quad (13)$$

$$c_w X_{N-1,k} + p_w X_{N,k-1} - s_w X_{N,k} = 0; 0 < k < N \quad (14)$$

$$p_w X_{N,N-1} - \gamma X_{N,N} = 0 \quad (15)$$

where  $r_w = \theta + c_w$ ,  $s_w = \theta + p_w$  and  $t_w = \theta + c_w + p_w$  are introduced to simplify the text, and  $X_{j,k}$  denotes the value of  $x_{j,k}(t)$  at equilibrium, namely, the number of peers in group  $(j, k)$  in steady-state.

From (9)-(15), the number of peers in each group  $(j, k)$  can be obtained as

$$X_{0,0} = \frac{\lambda}{r_w} \quad (16)$$

$$X_{j,0} = \frac{c_w}{t_w} X_{j-1,0}; 0 < j < N \quad (17)$$

$$X_{j,k} = \frac{1}{t_w} [c_w X_{j-1,k} + p_w X_{j,k-1}]; \quad (18)$$

$$0 < j < N, 0 < k < j.$$

$$X_{j,j} = \frac{p_w}{r_w} X_{j,j-1}; 0 < j < N \quad (19)$$

$$X_{N,0} = \frac{c_w}{s_w} X_{N-1,0} \quad (20)$$

$$X_{N,k} = \frac{1}{s_w} [c_w X_{N-1,k} + p_w X_{N,k-1}]; 0 < k < N \quad (21)$$

$$X_{N,N} = \frac{p_w}{\gamma} X_{N,N-1}. \quad (22)$$

Since (16)-(22) is a lower-triangular system, it can be easily solved by using forward-substitution.

For the sake of simplifying subsequent expressions, we refer to peers that are downloading window  $j$ , regardless of their playing window, as *peers in group  $j$* ; so, depending on the circumstances, we will speak about group  $j$  or group  $(j, k)$ . The number of peers in group  $j$  in steady-state is denoted by  $X_j$  and can be calculated as

$$X_j = \sum_{k=0}^j X_{j,k}. \quad (23)$$

In addition, we denote the total number of peers by  $X$  and it can be obtained as

$$X = \sum_{j=0}^N X_j. \quad (24)$$

Define  $V_j$  as the number of peers in steady-state whose downloading-window is less than or equal to  $J \in [0 : N - 1]$ , that is

$$V_J = \sum_{j=0}^J \sum_{k=0}^j X_{j,k} = \sum_{j=0}^J X_j. \quad (25)$$

Note that  $V_J$  can also be interpreted as the number of leechers that can be served by peers in group  $J + 1$ . As we will see in the following section, such an interpretation is very useful when defining resource allocation schemes.

### 3.3. Abundance conditions and Uniform Allocation

In this section, we consider a simple scheme, where resources from peers in group  $J$  are uniformly distributed among all the peers downloading previous windows (cf. (Baez-Esquivel et al., 2013)). As a result of this scheme, which we refer to as Uniform Allocation (UA), leechers in group  $(j, k)$  receive a fraction  $\frac{c_w X_j}{V_{j-1}}$  from the total data rate provided by peers in group  $J$  (i.e.,  $\mu_w X_j$ ). Considering this, the upload data rate from other peers, allocated to leechers in group  $(j, k)$  is given by

$$U_{leech}(j, k) = \sum_{J=j+1}^N \frac{X_{j,k}}{V_{J-1}} \mu_w X_J. \quad (26)$$

Note that (26) does not mean that the resources from one peer at group  $J$  are divided into all the leechers at lower groups; what this equation indicates is that the resources from the whole population of peers in group  $J$  are uniformly allocated among the leechers in lower groups.

In the context of a hybrid CDN-P2P system, we assume that the content-servers contribute with the upload data rate  $\nu$ , and that the UA strategy is also used to allocate these extra resources. Therefore, the total upload data rate assigned to group  $(j, k)$  in steady-state can be expressed as

$$U_{ud}(j, k) = X_{j,k} \left[ \mu_w \sum_{J=j+1}^N \frac{X_J}{V_{J-1}} + \frac{\nu}{V_{N-1}} \right], \quad (27)$$

and the abundance conditions are guaranteed if

$$c_w X_{j,k} \leq X_{j,k} \left[ \mu_w \sum_{J=j+1}^N \frac{X_J}{V_{J-1}} + \frac{\nu}{V_{N-1}} \right], \quad (28)$$

for  $j \in [0 : N - 1]$  and  $k \in [0 : j]$ , see (8). This inequation can be reduced to

$$c_w \leq \sum_{J=j+1}^N \mu_w \frac{X_J}{V_{J-1}} + \frac{\nu}{V_{N-1}}, \quad (29)$$

for  $j \in [0 : N - 1]$ . By substituting (16)-(22) in (29), abundance conditions can be explicitly given in terms of the network parameters.

Several strategies may be implemented to guarantee abundances conditions. For example, the service provider could reserve enough upload data rate from the CDN ( $\nu$ ) in order to satisfy (29) for any  $j$ , given the remaining network parameters. In Section 4 we propose a scheme to facilitate operating in abundance.

## 4. Prioritized-windows allocation scheme

The UA scheme asymmetrically distribute resources among peers, since leechers in the first few windows, which can access plenty of resources from other leechers, may also be served by seeders or servers; while leechers in the last few windows may be experiencing penury.

The *Prioritized Windows Allocation* (PWA) scheme (Torres-Cruz et al., 2017) avoids this asymmetry by assigning a priority to the peers depending on the window they are currently downloading. Thus, peers downloading higher index windows have a higher priority to access

resources from servers. In this section we propose a *generalized* version of PWA (GPWA), where prioritization is used to distribute not only resources from servers but also resources from peers.

#### 4.1. GPWA description

In GPWA, the upload data rate from peers in group  $J$ , (i.e.,  $\mu_w X_J$ ), is distributed among peers in groups  $j < J$ . The fraction of  $\mu_w X_J$  that is allocated to group  $j$  is proportional to  $X_j(j+1)^\epsilon$ , where  $\epsilon \geq 0$  is a tuning parameter that allows us to control the priority level that a window receives. The fraction of  $\mu_w X_J$  allocated to group  $j$ , denoted by  $U_j(j)$ , can be defined as

$$U_j(j) = u_w X_J \frac{X_j(j+1)^\epsilon}{V_{j-1}^*}, \quad (30)$$

where  $V_{j-1}^*$  is given by

$$V_{j-1}^* = \sum_{m=0}^{j-1} X_m(m+1)^\epsilon. \quad (31)$$

Note that  $V_{j-1}^*$  is required to guarantee that  $\sum_j U_j(j) = u_w X_J$  and can be interpreted as a weighted sum of leechers that can access resources from group  $J$ .

As mentioned,  $\epsilon$  can be used to tune the priorities assigned to the peers. E.g., if  $\epsilon = 0$ , GPWA turns into UA, i.e., no prioritized windows exist, and if  $\epsilon \rightarrow \infty$ ,  $U_j(j-1) \rightarrow u_w X_J$  and  $U_j(j) \rightarrow 0$  for  $j < J-1$ , then resources from group  $J$  will be exclusively allocated to group  $J-1$ .  $\epsilon$  can be used to optimize different performance metrics, and in Section 4.3 we focus on tuning it to minimize the upload data rate from the CDN.

From (30), (31), and recalling that resources from servers are also allocated according to GPWA, we can express the total upload data rate allocated to group  $j$  in steady-state as

$$U_{gpwa}(j) = X_j(j+1)^\epsilon \left( \sum_{J=j+1}^N \frac{\mu_w X_J}{V_{j-1}^*} + \frac{v}{V_{N-1}^*} \right), \quad (32)$$

and the abundance conditions are given by

$$c_w \leq (j+1)^\epsilon \left( \sum_{J=j+1}^N \frac{\mu_w X_J}{V_{j-1}^*} + \frac{v}{V_{N-1}^*} \right), \quad (33)$$

for  $j \in [0 : N-1]$ . Note that this implies that (16)-(22) are valid only if (33) holds for all  $j$ .

#### 4.2. Minimum upload data rate from the CDN

Since we are not addressing the problem of improving the cooperation level from peers, we assume that parameters modeling such a cooperation (i.e., leechers and seeders departure rates) are fixed. Additionally, we assume that in the hybrid network, resources from the CDN are allocated only if the system runs out of resources from peers. With these assumptions we can estimate how much upload data rate must be provided by the CDN to satisfy abundance conditions in the GPWA scheme.

From (33), we can re-write the abundance conditions as

$$v \geq v_j^{min} = \max \left\{ 0, V_{N-1}^* \left[ \frac{c_w}{(j+1)^\epsilon} - \sum_{J=j+1}^N \frac{\mu_w X_J}{V_{j-1}^*} \right] \right\}, \quad (34)$$

for  $j \in [0 : N-1]$ . The right side of this inequation can be interpreted as the minimum value of  $v$  that satisfies abundance for group  $j$ , and is denoted by  $v_j^{min}$ . The  $\max\{\cdot, \cdot\}$  function is introduced to impede  $v_j^{min}$  of potentially being negative when a large upload data rate from peers exists. The minimum upload data rate from the CDN that satisfies abundance in the whole system,  $v^{min}$ , is given by

$$v^{min} = \max_j \left\{ v_j^{min} \right\} \quad (35)$$

Notice that  $v^{min}$  is in terms of the number of peers in steady-state; however, during the system operation, this parameter may suffer variations, since the number of peers is not a constant. According to this, the exact amount of resources from the CDN to satisfy abundance is a dynamic parameter; therefore, it could be provided through Cloud-based CDN, as suggested in Haghighi et al. (2018). This approximation permits dynamically using bandwidth, virtual machines, and storage from programmable, general purpose hardware in distributed cloud sites; hence reducing the costs to maintain abundance. Despite this dynamism, (35) represents a very useful expression to estimate the scale of the required resources from the CDN. The previous arguments are also valid for the abundance conditions in terms of  $v^{min}$  that we present in the remaining of this paper.

#### 4.3. Heuristic for selecting $\epsilon$

We now describe a simple yet effective heuristic for selecting  $\epsilon$  in order to minimize  $v^{min}$ . Recall that choosing  $\epsilon = 0$  makes GPWA behave exactly like UA, i.e., a small fraction of the resources from seeders are assigned to leechers in group  $N-1$ , creating the need of a strong support from the CDN. On the other hand, if an excessively large value of  $\epsilon$  is selected, leechers in low index windows have limited access to resources from other leechers, hence, they require a lot of server bandwidth. Fig. 3 illustrates such requirements in terms of  $\epsilon$ . Particularly when  $\epsilon = 0$  and  $\epsilon = 2$ ,  $v_j^{min}$  reaches high values at the expected critical windows ( $j = N-1$  and  $j = 0$ , respectively). Fig. 3 also shows that intermediate values of  $\epsilon$  are better options to minimize the value of  $v^{min}$  (e.g.,  $\epsilon = 1$ ).

To intuit our heuristic, observe that  $v^{min}$  would be minimal if  $v_j^{min}$  is constant for all  $j$ , i.e., the upload data rate from peers is perfectly distributed among all the leechers. Note that  $v_j^{min}$  is a constant if the difference  $n(j) = v_{j+1}^{min} - v_j^{min} = 0$  for  $j \in [0 : N-2]$ . In order to analyze  $n(j)$  for small values of  $j$ , in this section we omit the *max* operation from the definition of  $v_j^{min}$  (cf. in Fig. 3 where negative values of  $v_j^{min}$  are shown).

As a result of the GPWA definition, the possible shapes of  $v_j^{min}$  are restricted and it is not possible to achieve a constant behavior by simply selecting the value of  $\epsilon$ ; however, a good approximation can be obtained by fixing  $n(0) = 0$ . Therefore, we select  $\epsilon$  in order to satisfy

$$n(0) = v_1^{min} - v_0^{min} = 0 \quad (36)$$

By substituting (35) in (36), we obtain

$$\left[ c_w - \sum_{J=1}^N \frac{\mu_w X_J}{V_{j+1}^*} \right] - \left[ \frac{c_w}{2^\epsilon} - \sum_{J=2}^N \frac{\mu_w X_J}{V_{j-1}^*} \right] = 0, \quad (37)$$

which can be reduced to

$$c_w - \frac{c_w}{2^\epsilon} - \frac{\mu_w X_1}{V_0^*} = 0. \quad (38)$$

According to (31),  $V_0^* = X_0$ . In addition, from (16), (17) and (19) we can obtain  $X_0 = X_{0,0}$  and  $X_1 = X_{1,0} + X_{1,1}$  as

$$X_0 = \frac{\lambda}{r_w} \quad (39)$$

$$X_1 = \frac{\lambda c_w}{r_w t_w} \left( 1 + \frac{p_w}{r_w} \right) \quad (40)$$

Then, by substituting (39) and (40) in (38), we find an approximated value of  $\epsilon_{opt}$ , denoted by  $\epsilon_{app}$ , as

$$\epsilon_{app} = \log_2 \left( \frac{r_w}{r_w - \mu_w} \right) \quad (41)$$

In Fig. 3, we also show  $v_j^{min}(\epsilon_{opt})$  (obtained through exhaustive evaluation) and we compared it with  $v_j^{min}(\epsilon_{app})$ . Note that  $v^{min}(\epsilon_{opt}) = 1.28$

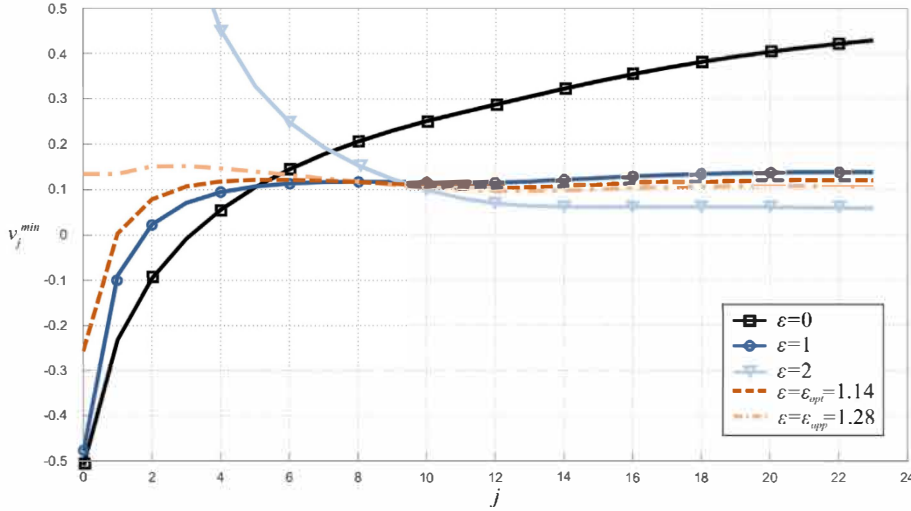


Fig. 3. Normalized, minimum upload bandwidth required to achieve abundance in window  $j$  for  $N = 24$ ,  $S = 12$ ,  $R = 8$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.06$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.002035$ .

and  $v_j^{\min}(\epsilon_{app}) = 1.14$ , i.e., similar values.

In Section 10, numerical evaluations support our claim that the approximated solution provides a way to distribute the resources of the peers that effectively reduces the upload data rate required from the CDN to satisfy abundance conditions.

## 5. Immediate-neighbors Uniform Allocation scheme

In this section, we briefly describe a resource allocation scheme proposed by Zhao et al. (2016), and evaluate its performance using our framework. Our propose is twofold. First, to show that our framework is expressive enough to capture a variety of schemes, and to provide a comparative performance of GPWA against a representative of the state-of-the-art.

In their paper, Zhao et al. analyze a VoD system and propose a peer-selection strategy to share content. In this strategy, every user randomly selects its serving peers from the first  $M = QX$  peers with superior downloading progress (immediate upper neighbors), where  $0 < Q < \bar{Q} = X_N/X$ . We refer to this scheme as Immediate-Neighbors Uniform Allocation (INUA). To evaluate the INUA scheme with our framework, we model its behavior in terms of peer populations.

Since every peer in window  $j$  has the same downloading progress, all of them select their serving peers from the same set of windows. Now, let  $\mathcal{U}_j$  be this set of windows and  $M_j = |\mathcal{U}_j|$  be its cardinality, then,  $\mathcal{U}_j = \{j+1, \dots, j+M_j\}$ , where  $M_j$  must satisfy that

$$\sum_{m=j+1}^{j+M_j-1} X_m < XQ \leq \sum_{m=j+1}^{j+M_j} X_m. \quad (42)$$

Since each peer randomly selects its serving peers from all the windows in  $\mathcal{U}_j$ , the number of peers in group  $j$  that are served by a window  $J \in \mathcal{U}_j$  is given by

$$Y_J(j) = X_j \frac{X_J}{\sum_{m \in \mathcal{U}_j} X_m}, \quad (43)$$

and, as a consequence, the upload bandwidth from group  $J$  that is assigned to peers in group  $j$  is  $\mu_w X_J \frac{Y_J(j)}{\sum_{n \in \mathcal{D}_j} Y_J(n)}$ ; where  $\mathcal{D}_j$  is the set of windows that are served by group  $J$  and it is implicitly defined in (42). Then, by assuming a pure P2P network, the upload bandwidth assigned to peers in group  $j$  can be written as

$$U_{inua}^j = \sum_{J \in \mathcal{U}_j} \mu_w X_J \frac{Y_J(j)}{\sum_{n \in \mathcal{D}_J} Y_J(n)}, \quad (44)$$

As in previous analysis, we consider that resources from servers may be required in order to achieve abundance conditions; furthermore, we consider that such resources are uniformly distributed among peers in all the windows. We must point out that this policy is not explicit in Zhao et al. (2016); however, we consider that it represents the natural extension to INUA when adding resources from servers. Then, the servers' resources allocated to window  $j$  are  $v \frac{X_j}{V_{N-1}}$ . By repeating the analysis developed in Subsection 4.2, we obtain the abundance condition for window  $j$  as

$$v \geq v_j^{\min} = \max \left\{ 0, V_{N-1} \left[ c_w - U_{inua}^j \right] \right\}. \quad (45)$$

Lastly, the minimum upload bandwidth required from the servers to satisfy the abundance condition can be obtained through (35).

Notice that the previous analysis illustrates how to incorporate the effects of INUA in our framework. Moreover, in Section 10 we present numerical evaluations of this scheme and it is compared with GPWA.

## 6. Tridimensional classification with download restrictions

In VoD services, users applications prefetch content as a way of reducing the frequency and duration of stalling events; however, unrestricted prefetching of content has the disadvantage that the non-played buffer content will be discarded if the user aborts the video playing (Ramos-Muñoz et al., 2014). For this reason, VoD applications employ *download restrictions* (DR), i.e. pause the downloading process if the size of the non-played buffer becomes larger than a threshold  $S$  (i.e., excessive pre-buffering), and resumes downloading if the non-played buffer becomes smaller than a second threshold  $R < S$ . In this section, we propose an extension to our basic model which enables us to analyze the more practical scenario where devices (e.g., mobile) have these download restrictions.

In order to incorporate DR into our model, we assume that thresholds  $S$  and  $R$  are given in term of numbers of windows and  $S \leq N$ . We include the binary classification parameter  $h$ , for which  $h = 0$  when the peer is allowed to download or  $h = 1$  when downloading is paused due to excessive pre-buffering. Accordingly, peers are classified into groups that are identified by a tridimensional label  $(h, j, k)$ . We use  $X_{j,k}^h$  to denote the number of peers in steady-state in each of these groups. In this model,  $j$  and  $k$  have, in general, the same meaning as in the basic model; but for groups  $(1, j, k)$ ,  $j$  indicates that window  $j - 1$  was completely downloaded. In Fig. 4, we represent the tridimensional classification with one plane for every value of  $h$ .



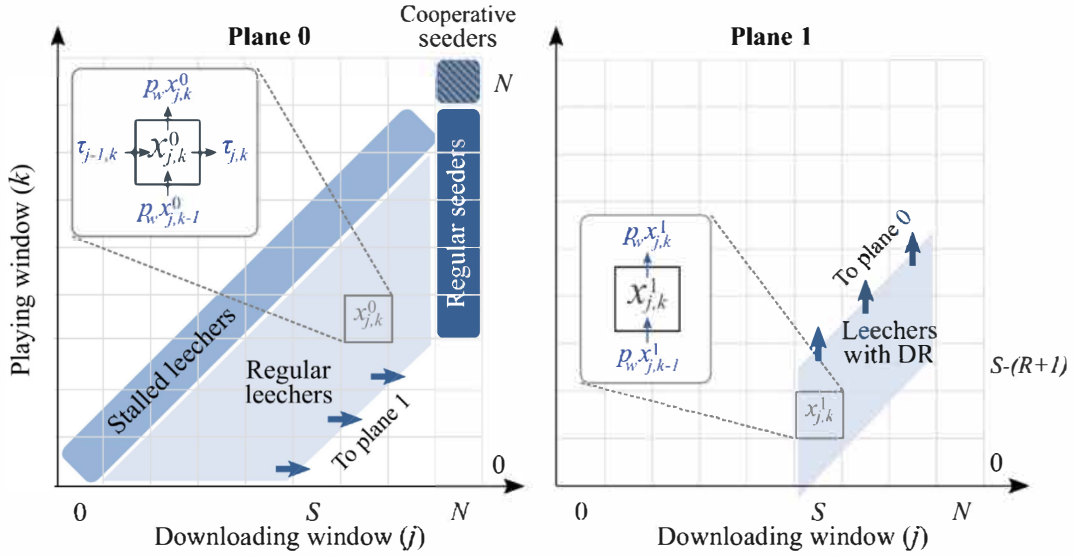


Fig. 4. Peer's classification in the scenario with DR.

### 6.1. Fluid model with download restrictions

In addition to the transitions of the basic model, described in 3.2, for the tridimensional model we also consider the following transitions:

- Since at the beginning peers arrive at the system with an empty buffer, they are initially classified in group  $(0, 0, 0)$ .
- Peers in group  $(0, j, k)$  that satisfy  $S - 1 \leq j < N$  and  $k = j - (S - 1)$  have accumulated  $S - 1$  non-played windows; hence, if they finish the downloading of an additional window, they will transit to group  $(1, j + 1, k)$ , rather than to group  $(0, j + 1, k)$ . Therefore, these peers transit from plane  $h = 0$  to plane  $h = 1$ . As a consequence, for  $S \leq j < N$  and  $k < j - (S - 1)$  as well as  $j = N$  and  $k < N - S$ , we have  $X_{j,k}^0 = 0$ .
- Since peers in plane  $h = 1$  are not allowed to download additional windows, they can not increase  $j$ ; hence, they transit to other groups only by increasing  $k$ .
- Peers in plane  $h = 0$  return to plane  $h = 1$  when the number of non-played windows becomes  $R$ . In other words, peers in group  $(1, j, k)$  transits towards group  $(0, j, k + 1)$  when  $k = j - (R + 1)$ .

Under these considerations, the steady state of the fluid model in abundance can be obtained through the following set of equations:

$$\lambda - r_w X_{0,0}^0 = 0 \quad (46)$$

$$c_w X_{j-1,0}^0 - t_w X_{j,0}^0 = 0; 1 \leq j < S \quad (47)$$

$$c_w X_{j-1,k}^0 + p_w X_{j,k-1}^0 - t_w X_{j,k}^0 = 0; \quad (48)$$

$$1 < j < S, 0 < k < j$$

$$p_w X_{j,j-1}^0 - r_w X_{j,j}^0 = 0; 1 \leq j < N \quad (49)$$

$$c_w X_{j-1,j-S+1}^0 - t_w X_{j,j-S+1}^0 = 0; S \leq j < N \quad (50)$$

$$c_w X_{j-1,k}^0 + p_w X_{j,k-1}^0 - t_w X_{j,k}^0 = 0; \quad (51)$$

$$S \leq j < N, j - S + 1 < k < j, k \neq j - R$$

$$c_w X_{j-1,j-R}^0 + p_w (X_{j,j-R-1}^0 + X_{j,j-R-1}^1) \quad (52)$$

$$-t_w X_{j,j-R}^0 = 0; S \leq j < N$$

$$c_w X_{j-1,j-S}^0 - s_w X_{j,j-S}^1 = 0; S \leq j < N \quad (53)$$

$$p_w X_{j,k-1}^1 - s_w X_{j,k}^1 = 0; \quad (54)$$

$$S \leq j < N, j - S < k < j - R;$$

$$c_w X_{N-1,N-S}^0 - s_w X_{N,N-S}^0 = 0; \quad (55)$$

$$c_w X_{N-1,k}^0 + p_w X_{N,k-1}^0 - s_w X_{N,k}^0 = 0; \quad (56)$$

$$N - S < k < N$$

$$p_w X_{N,N-1}^0 - \gamma X_{N,N}^0 = 0. \quad (57)$$

Note that (46)-(49), (55)-(57) are analogous to (9)-(15); whereas (50)-(54) represent flows of peers that do not exist in the basic-bidimensional model. This set of equations is also a lower-triangular system, hence, we also solved it by using forward-substitution.

Note that the limits for  $j$  and  $k$  in (46)-(57) are valid for  $S < N$ . If  $S = N$  then there are no download restrictions and the system should be analyzed with the basic model.

### 6.2. Abundance condition, download restrictions, and GPWA

We now focus on determining the required upload data rate from the CDN to guarantee the abundance condition under the GPWA scheme. First define  $X_j^0$  and  $X_j^1$  as the number of peers in planes 0 and 1, respectively, that have downloaded window 0 to window  $j - 1$ . These values can be computed as

$$X_j^h = \sum_{k=0}^j X_{j,k}^h, \quad (58)$$

for  $h \in [0, 1]$ . The weighted sum of leechers that can access resources from group  $J$  is defined as

$$V_{J-1}^{0*} = \sum_{m=0}^{J-1} X_m^0 (m+1)^\epsilon. \quad (59)$$

Note that this sum does not include leechers in plane 1 because they are not consuming download resources.

By using the definition of GPWA, i.e. (32), the upload data rate allocated to group  $j$  can be computed as

$$U_{gpwa}^{DR}(j) = X_j^0(j+1)^\epsilon \left( \sum_{j=j+1}^N \frac{\mu_w(X_j^0 + X_j^1)}{V_{j-1}^{0*}} + \frac{v}{V_{N-1}^{0*}} \right). \quad (60)$$

for  $j \in [0 : N - 1]$ .

Since abundance conditions for group  $j$  are satisfied when  $c_w X_j^0 \leq U_{gpwa}^{DR}$ , we can determine the minimum upload data rate, from the CDN, required to maintain abundance as

$$v_j^{min} = \max \left\{ 0, V_{N-1}^{0*} \left[ \frac{c_w}{(j+1)^\epsilon} - \sum_{j=j+1}^N \frac{\mu_w(X_j^0 + X_j^1)}{V_{j-1}^{0*}} \right] \right\} \quad (61)$$

for  $j \in [0 : N - 1]$ . Therefore, to maintain abundance in the whole system, the upload data rate must be greater than or equal to  $v^{min} = \max_j \{v_j^{min}\}$ . Last, since  $X_0^0$  and  $X_1^0$  can also be expressed by (39) and (40), we can also select  $\epsilon$  according to the criteria described in Section 4.3.

## 7. Tridimensional classification with interaction functionalities

In this section we further extend our model to consider that users can interact with the video playback through what is known as VCR functions (Choi et al., 2012). Such a general model can capture the effects of forward and backward random-seeks (i.e., instantaneous jumps in the playback position). Since introducing intentional pauses is equivalent to increasing the playback time (cf. (Zhao et al., 2013)), our model could describe the effect of such pauses by reducing the playback rate  $p_w$ . We refer to this scenario as with *Interaction Functionalities* (IF). For clarity and simplicity, in this section we do not consider download restrictions.

One of the main challenges introduced by interaction functionalities is that users could download or play the video in random order, thus invalidating the assumption that a peer downloading window  $j$  necessarily has buffered all the previous windows. This can significantly increase the complexity of the system analysis as well as the implementation of resource allocation schemes.

Thus, we propose implementing resource allocation schemes that only considers the last set of sequentially downloaded windows by a peer. Meaning peers will only be able to supply sets of windows with continuous indexes from  $i$  to  $j - 1$ . Note that  $i = 0$  for new peers and that it changes only after a random-seek outside the buffered content, and recall that  $j$  is the current downloading window. While such a consideration reduces the amount of available resources from peers, the consideration also simplifies the state information at the tracker because it only has to maintain the values of  $i$  and  $j$ , rather than the whole map of downloaded windows. Furthermore, we can classify peers in terms of three window-indexes,  $(i, j, k)$  and we can establish the relations among groups in our fluid model.

### 7.1. Fluid model with interaction functionalities

In this subsection, group  $(i, j, k)$  is composed of peers that resumed the downloading process in window  $i$ , are downloading window  $j$  (hence  $j \leq i$ ) and are playing window  $k$ , for  $k < j$ . Similar to previous model variations,  $k = j$  is used to classify stalled leechers; therefore,  $k \in [i : j]$ .

We use the tridimensional representation to depict how peers transit between groups, i.e., each plane corresponds to a value of  $i$ . Fig. 5 shows the representation of an arbitrary plane  $i \neq 0$ . Notice that plane  $i = 0$  corresponds to the groups defined in the basic scenario (see Fig. 2).

In addition to the transitions defined in the basic model, in the scenario with IF a peer can also transit from one group to another as a

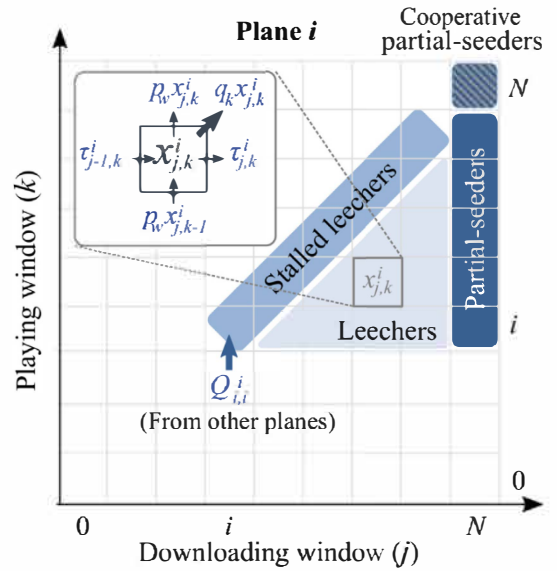


Fig. 5. Peer's classification in the scenario with IF.

result of a random forward (backward) seek. A peer in group  $(i, j, k)$  that jumps forward (backward) to window  $l$  is re-classified as  $(i, j, l)$ , if  $l \in [i : j]$ ; since the peer jumps to a buffered window and only its playback classification is affected. If  $l \notin [i : j]$ , the peer halts playing window  $k$  and downloading window  $j$ , and becomes a stalled leecher downloading window  $l$ ; since the set of continuously buffered windows for the peer is reduced to  $l$  the peer transits to plane  $l$  and is re-classified as  $(l, l, l)$ . As a consequence of such a change of planes, peers will permanently lose its previously buffered windows; therefore, seeders in planes  $i > 0$  do not store the whole video and are named *partial-seeders* (see Fig. 5).

In order to integrate the transitions resulting from random-seeks, we denote by  $q_{k,l}$  the rate at which users playing window  $k$  jump to window  $l$ , for  $k \in [0 : N]$  and  $l \in [0 : N - 1]$ . Then, the rate at which users playing window  $k$  jump to any other window,  $q_k$ , is given by

$$q_k = \sum_{n=0}^{N-1} q_{k,n}. \quad (62)$$

The probability distribution of  $q_{k,l}$  can be obtained through field measurements, cf. (Chen et al., 2014).

Now, let  $Q_{j,k}^i$  be the rate of peers that arrive at group  $(i, j, k)$  as a result of random-seeks. It is given by

$$Q_{j,k}^i = \sum_{\substack{m=i \\ m \neq k}}^j q_{m,k} X_{j,m}^i, \quad (63)$$

if  $i \leq k \leq j$ , because in this case, only inter-plane random-seeks contribute to this rate. Note that  $Q_{i,i}^i$  can be interpreted as the rate of peers arriving at plane  $i$  from groups where window  $i$  is not buffered and is given by

$$Q_{i,i}^i = \sum_{l=0}^{i-1} \sum_{j=l}^{i-1} \sum_{k=l}^j q_{k,i} X_{j,k}^l + \sum_{l=i+1}^N \sum_{j=l}^{N-1} \sum_{k=l}^j q_{k,i} X_{j,k}^l, \quad (64)$$

where the first term is the contribution from peers in a plane  $l < i$  that still have not downloaded window  $i$ , whereas the second is the contributions from peers in a plane  $l > i$ .

By incorporating (62)-(64) in the fluid model under abundance conditions and in steady-state, we can establish the following set of equations:

$$\lambda + Q_{0,0}^0 - (r_w + q_0) X_{0,0}^0 = 0 \quad (65)$$

$$Q_{i,i}^i - (r_w + q_i)X_{i,i}^i = 0; 1 \leq i \leq N-1 \quad (66)$$

$$c_w X_{j-1,i}^i + Q_{j,i}^i - (t_w + q_i)X_{j,i}^i = 0; \quad (67)$$

$$0 \leq i \leq N-2, i < j < N$$

$$c_w X_{j-1,i}^i + p_w X_{j,k-1}^i + Q_{j,k}^i - (t_w + q_k)X_{j,k}^i = 0; \quad (68)$$

$$0 \leq i \leq N-3, i < j < N, i < k < j$$

$$p_w X_{j,j-1}^i + Q_{j,j}^i - (r_w + q_j)X_{j,j}^i = 0; \quad (69)$$

$$0 \leq i \leq N-2, i < j < N$$

$$c_w X_{N-1,i}^i + Q_{N,i}^i - (s_w + q_i)X_{N,i}^i = 0; \quad (70)$$

$$0 \leq i \leq N-1$$

$$c_w X_{N-1,k}^i + p_w X_{N,k-1}^i + Q_{N,k}^i - (s_w + q_k)X_{N,k}^i = 0; \quad (71)$$

$$0 \leq i \leq N-2, i < k < N$$

$$p_w X_{N,N-1}^i - (\gamma + q_N)X_{N,N}^i = 0; 0 \leq i \leq N-1 \quad (72)$$

where,  $X_{j,k}^i$  is the number of peers in group  $(i, j, k)$ .

Observe that (65) and (66) are analogous to (9), except that in the former equations we include transitions rates resulting from random-seeks, i.e., we add  $Q_{j,k}^i$  and subtract  $q_k X_{j,k}^i$ . The same is true when comparing (67)-(72) with (10)-(15), respectively. If  $q_{k,l} = 0$ , for all  $k, l$ , the fluid model with IF becomes the basic model.

Unlike previous extensions, the system of equations for the scenario with IF cannot be solved by simply using forward-substitution. However, we solve this system through LU factorization with partial pivoting, for  $N \leq 24$  (see Section 10). For larger values of  $N$ , iterative methods are recommended (e.g., Gauss-Seidel).

## 7.2. Abundance conditions with IF and GPWA

Let  $X_j^i$  be the number of peers in plane  $i$  downloading window  $j$ , and  $X_j$  be the total number of peers downloading this same window. These variables can be expressed as

$$X_j^i = \sum_{k=i}^j X_{j,k}^i, \quad (73)$$

$$X_j = \sum_{i=0}^{N-1} X_j^i. \quad (74)$$

Additionally, let  $V_{j-1}^{I*}$  be the weighted sum of leechers that can be served by peers in possession of windows  $I$  to  $J-1$ . Particularly, the weighted sum of this kind of leechers in plane  $n$  is given by  $\sum_{m=l}^{J-1} X_m^n (m+1)^\epsilon$ ; therefore, by summing over all the planes, we have

$$V_{j-1}^{I*} = \sum_{n=0}^{N-1} \sum_{m=l}^{J-1} X_m^n (m+1)^\epsilon. \quad (75)$$

Similar to previous analyses, the ratio  $X_j(j+1)^\epsilon / V_{j-1}^{I*}$  represents the proportion of upload data rate from  $\mu_w X_j^I$  that is allocated to peers downloading window  $j$ . Hence, by assuming that resources from both

**Table 1**

Markov chain transitions for the scenario with DR.

Trans. Rate	New state	Restrictions
$\lambda$	$\{\dots, Y_{0,0}^0 + 1, \dots\}$	$-$
$\theta Y_{j,k}^h$	$\{\dots, Y_{j,k}^h - 1, \dots\}$	$\forall (h, j, k)$ , except $(0, N, N)$
$\gamma Y_{N,N}^0$	$\{\dots, Y_{N,N}^0 - 1, \dots\}$	$-$
$\tau_{j,k}^0$	$\{\dots, Y_{j,k}^0 - 1, Y_{j+1,k}^0 + 1, \dots\}$	$0 \leq j < S \text{ \& } 0 \leq k \leq j$ $S \leq j < N \text{ \& } j - S < k \leq j$
$\tau_{j,j-(S-1)}^0$	$\{\dots, Y_{j,j-S+1}^0 - 1, Y_{j+1,j-S+1}^0 + 1, \dots\}$	$S - 1 \leq j < N$
$p_w Y_{j,k}^0$	$\{\dots, Y_{j,k}^0 - 1, Y_{j,k+1}^0 + 1, \dots\}$	$1 \leq j < S \text{ \& } 0 \leq k < j$ $S \leq j < N \text{ \& } j - S + 1 \leq k < j$ $j = N \text{ \& } j - S \leq k < N$
$p_w Y_{j,k}^1$	$\{\dots, Y_{j,k}^1 - 1, Y_{j,k+1}^1 + 1, \dots\}$	$S \leq j < N \text{ \& } j - S \leq k < j - R - 1$
$p_w Y_{j,j-R-1}^1$	$\{\dots, Y_{j,j-R-1}^1 - 1, Y_{j,j-R}^1 + 1, \dots\}$	$S \leq j < N$

peers and the CDN are allocated according to the GPWA, the total upload data rate assigned to download window  $j$  is

$$U_{gpwa}^{IF}(j) = X_j(j+1)^\epsilon \left( \sum_{I=0}^j \sum_{J=I+1}^N \frac{\mu_w X_J^I}{V_{J-1}^{I*}} + \frac{v}{V_{N-1}^{0*}} \right). \quad (76)$$

Recall that abundance holds if  $c_w X_j \leq U_{gpwa}^{IF}(j)$ . Thus, we can express the minimum upload data rate from the CDN that satisfies this condition as

$$v_j^{min} = \max \left\{ 0, V_{N-1}^{0*} \left[ \frac{c_w}{(j+1)^\epsilon} - \sum_{I=0}^j \sum_{J=I+1}^N \frac{\mu_w X_J^I}{V_{J-1}^{I*}} \right] \right\} \quad (77)$$

## 8. Markov chains associated to fluid models

In order to validate our fluid models, as in [Baez-Esquivel et al. \(2013\)](#), [Torres-Cruz et al. \(2017\)](#) and [Rivero-Angeles and Rubino \(2010\)](#), we develop Markov-chain models for the scenarios of interest. Markov chain models have the advantage of providing performance results even in penury conditions, but, unlike the fluid models, have the disadvantage that, except in very special cases, we do not get closed-form expressions. For brevity, we only present the scenarios of the most practical interest, namely, those with DR and IF.

For the scenario with DR, we define a multidimensional Markov chain, whose state is a vector  $\mathbb{Y}$ , containing the elements  $Y_{j,k}^h(t) \geq 0$ ,  $h \in [0, 1]$ ,  $j \in [0 : N]$  and  $k \in [0 : j]$ , defined as the number of peers in group  $(h, j, k)$  at time  $t$ . We consider the standard ‘‘exponential assumptions’’ (plus independence); therefore,  $\mathbb{Y}$  is a continuous-time homogeneous Markov chain, and we also consider that its initial state is  $Y_{j,k}^0(0) = 0$ , for all  $(j, k)$ .

We use  $\{\dots, Y_{j,k}^h \pm 1, \dots\}$  to denote a vector containing the same elements as vector  $\mathbb{Y}$ , with the same values, but with the exception that the value of component  $Y_{j,k}^h(t)$  has been increased (or decreased) by one. This notation allows us to express transitions in the chain, e.g., we can write that the chain transits from state  $\mathbb{Y}$  to state  $\{Y_{0,0}^0 + 1, \dots\}$ , at rate  $\lambda$ , as a result of a new peer arrival.

All possible transitions from an arbitrary state  $\mathbb{Y}$ , and their corresponding rates are summarized in [Table 1](#). These transitions include the effects of arrivals and departures of peers and the downloading and playback processes.

Note that  $\tau_{j,k}^0$  depends on the resource allocation scheme and is given by  $\tau_{j,k}^0 = \min \left\{ c_w Y_{j,k}^0(t), \mathcal{U}(j, k) \right\}$ ; where  $\mathcal{U}(j, k)$  can be obtained by sub-

**Table 2**

Markov chain transitions for the scenario with IF.

Trans. rate	New state	Restrictions
$\lambda$	$\{\dots, Y_{0,\emptyset}^0 + 1, \dots\}$	–
$\theta Y_{j,k}^i$	$\{\dots, Y_{j,k}^i - 1, \dots\}$	$\forall (i,j,k)$ , except $(i,N,N)$
$\gamma Y_{N,N}^i$	$\{\dots, Y_{N,N}^i - 1, \dots\}$	–
$\tau_{j,k}^i$	$\{\dots, Y_{j,k}^i - 1, Y_{j+1,k}^i + 1, \dots\}$	$i \leq j < N$ & $i \leq k \leq j$
$p_w \gamma_{j,k}^i$	$\{\dots, Y_{j,k}^i - 1, Y_{j,k+1}^i + 1, \dots\}$	$i < j < N$ & $i \leq k < j$
$q_{k,l} Y_{j,k}^i$	$\{\dots, Y_{j,k}^i - 1, Y_{j,l}^i + 1, \dots\}$	$i \leq l \leq j$
$q_{k,l} Y_{j,k}^i$	$\{\dots, Y_{j,k}^i - 1, Y_{l,i}^i + 1, \dots\}$	$l < i$ or $l > j$

**Table 3**

Numerical values for the input parameters.

Parameter	Value
$\lambda$	0.04
$\theta$	0.006
$\gamma$	0.006
$c$	0.00407
$\mu$	0.00254
$p$	0.5c
$N$	24

stituting  $X_{j,k}^0$  by  $Y_{j,k}^0(t)$  in the equations that define the resource allocation schemes (e.g., (60) describes the GPWA scheme for the scenario with DR).

Table 2 defines the Markov chain associated with the model in Section 7 (i.e., IF). For the IF scenario, we make the same assumptions as in the case of the chain for the DR scenario. In this case,  $\mathbb{Y}$  must contain all the elements  $Y_{j,k}^i(t) \geq 0$  for  $i \in [0 : N - 1]$ ,  $j \in [i : N]$  and  $k \in [i : j]$ .

Lastly, it is important to mention that the mathematical relations between these Markov-chain models and the fluid models are out of the scope of this paper. In this regard, we recommend (Ethier and Kurtz, 1986), for some fundamentals, and (Aspirot et al., 2011), where the relations between such models are analyzed for the P2P system proposed in Rivero-Angeles and Rubino (2010).

## 9. Performance parameters

In this section we derive expressions for relevant network performance metrics from the solutions of the fluid model. We focus on the scenario with DR but an analogous analysis can be carried out for other scenarios.

The first performance metric we consider is *average network download data rate per window*  $j \in [0 : N - 1]$ , which is defined as

$$C_j = c_w X_j^0, \quad (78)$$

where  $X_j^0$  is given by (58). The average download data rate for the whole network ( $C$ ) is obtained by summing  $C_j \forall j$ .

Since content is supplied by both the CDN and the P2P network, it is relevant to calculate the amount of upload data rate provided by each. We assume that the primary source of content is the P2P network; hence, resources from the CDN are utilized only if the system runs out of resources from peers. Assuming GPWA is used, the upload data rate from peers to supply window  $j$  is given by

$$\mu_j = \min \left\{ c_w X_j^0, \mu_w X_j^0 (j+1)^\epsilon \sum_{J=j+1}^N \frac{(X_J^0 + X_J^1)}{V_{J-1}^{0*}} \right\}, \quad (79)$$

where  $X_j^1$  and  $V_{j-1}^{0*}$  are given by (58) and (59), respectively.

The  $\min\{\cdot, \cdot\}$  operation in (79) indicates if the demanded data rate for window  $j$  can be entirely supplied by peers or if the CDN must provide additional resources. In the latter case, the upload rate needed from the CDN to supply window  $j$  is

$$v_j^{DR} = c_w X_j^0 - \mu_j^{DR} \quad (80)$$

The entire upload data rate from peers ( $\mu$ ) and from the CDN ( $v$ ) can be obtained by summing (79) and (80), respectively, over  $j \in [0 : N - 1]$ .

We now emphasize the importance of the network parameters thus far considered. For instance, let us assume a given wireless P2P network which is complemented by a CDN infrastructure. In such scenario, (78) and (79) are directly related to the upload and download bandwidth required in the wireless network; moreover, (79) can be used to estimate the peers energy consumption to upload video fragments, which

is a cost that does not exist in traditional client-server networks. Additionally, (80) can be used with (61) to determine the cost of the CDN infrastructure required to supply the video of interest.

In order to evaluate QoS parameters, we calculate the proportion of peers experiencing stalling events at the different windows. Recall that the number of stalled leechers in each window is simply given by  $X_{jj}^0$ . Therefore, the total number of stalled leechers denoted by  $X_{st}$ , is computed as

$$X_{st} = \sum_{j=0}^{N-1} X_{jj}^0. \quad (81)$$

The total number of peers currently playing the video-file, denoted by  $X_{play}$ , is computed as

$$X_{play} = \sum_{j=0}^N (X_j^0 + X_j^1) - X_{N,N}^0. \quad (82)$$

Then, the proportion of stalled leechers,  $\alpha$ , is given by

$$\alpha = \frac{X_{st}}{X_{play}}, \quad (83)$$

and the probability distribution of the number of stalled leechers in terms of  $j$  is

$$\alpha_j = \frac{X_{jj}^0}{X_{st}}, \quad (84)$$

for  $j \in [0 : N - 1]$ . Notice that  $\alpha_0$  is the portion of leechers experiencing startup delay.

## 10. Numerical evaluation

Our numerical evaluations use the following statistics, which are reported in Ramos-Muñoz et al. (2014) and Dimopoulos et al. (2013), and correspond to YouTube services: average video-file length = 490 s, codification rate = 200kbits/s, chunk size = 64kBytes, average download rate = 400kbits/s and peer sojourn time = 172 s. From this data, we can directly obtain the following input parameters of the system: normalized download data rate  $c = 0.00407$  files/s, playback rate  $p = 0.5c$ , peer abandon rate  $\theta = \gamma = 1/172$  (we approximate it to 0.006) and maximum number of windows  $N = 192$ , though, for practical considerations, we use  $N = 24$  (8 chunks per window). To select the upload data rate, we consider that usually it is smaller than the download rate; hence, we use  $\mu = 0.00254$  files/s (equivalent to 250kbits/s). We arbitrarily choose  $\lambda = 0.04$ , but this parameter only scales the size of the peers' populations (because of the scalability property of P2P networks). In Table 3, we summarize the selected values.

### 10.1. Results for the DR scenario

Fig. 6 shows the number of peers in plane 0 for the scenario with DR. We observe that the numerical results from the fluid model and its associated Markov chain are highly consistent. In this figure we also



observe that there exist a high concentration of peers in groups  $(0, 0, 0)$  and  $(0, N, N)$ . This occurs because we consider that all peers arrive at the system with empty buffers and because peers tend to become seeders. As a result, the rest of the peers are dispersed in the remaining groups (including those in plane 1).

Figs. 7 and 8 show the minimum upload rate from servers that guarantee abundance conditions in the whole system ( $v^{min}$ ), for two levels of peers cooperation ( $\gamma = 0.006$  and  $\gamma = 0.6$ , respectively). In each case, we evaluate three resource-allocation schemes, namely, UA (Baez-Esquivel et al., 2013), INUA (Zhao et al., 2016) and GPWA. To obtain these results, the tuning parameter of GPWA,  $\epsilon$ , is selected according to the method described in Section 4.3; whereas, for the INUA scheme, we select  $Q = 0.001$  (a value that guarantees a high performance, according to exhaustive evaluations).

In these figures, we observe that small values of  $S$  significantly reduce  $v^{min}$ , for all the analyzed scenarios, because peers are allowed to download only the windows they are about to play, therefore, reducing the downloading demand. From this, we conclude that even though non-played windows can be useful to serve other peers, the load that they add to the network is larger than the benefits they may provide.

From Figs. 7 and 8, we also observe that the UA scheme is quite inefficient, because some resources are unnecessary allocated to peers in the first windows, as we discussed earlier. On the other hand, GPWA and INUA significantly reduce  $v^{min}$  and exhibit similar performances for large values of  $S$  and non-cooperative scenarios (Fig. 8). However, under cooperative scenarios (i.e., small values of  $\gamma$ ) GPWA achieves a more efficient use of the P2P-network resources, since it permits controlled allocation of upload bandwidth from seeders even to low windows, whereas in INUA this is not possible (only immediate neighbors

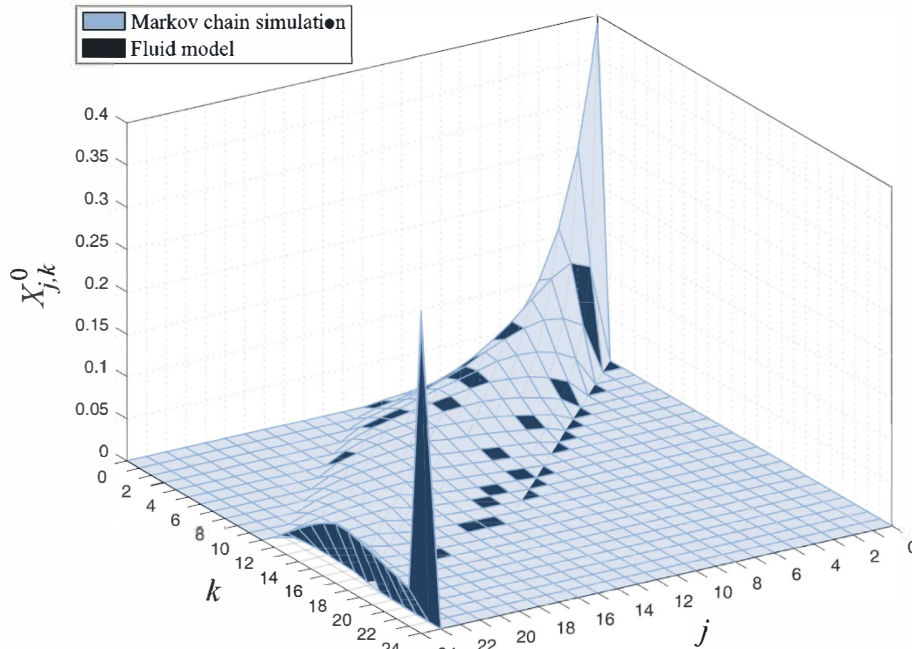


Fig. 6. Number of peers in plane 0 for every  $(j, k)$  group, considering  $N = 24$ ,  $S = 12$ ,  $R = 6$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.5c$ .

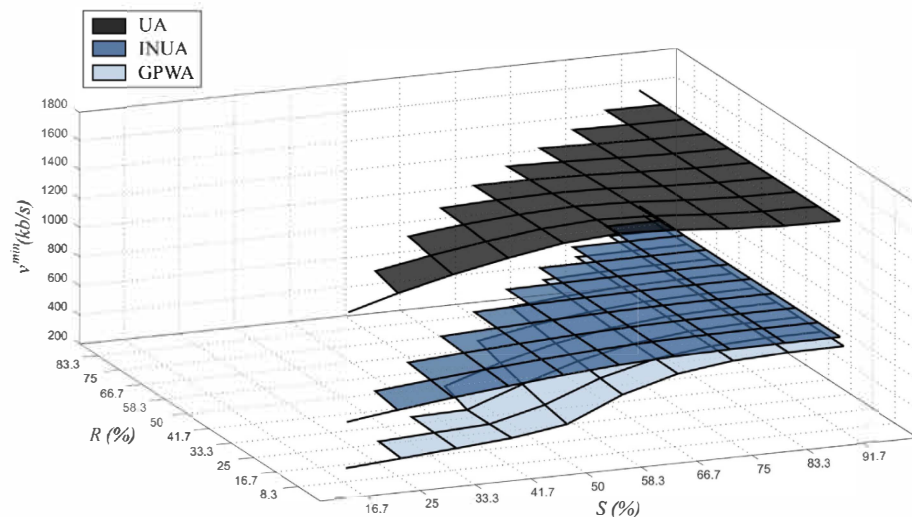


Fig. 7. Minimum upload rate required from servers to guarantee abundance conditions as a function of  $S$  and  $R$ ; for  $N = 24$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.5c$ .

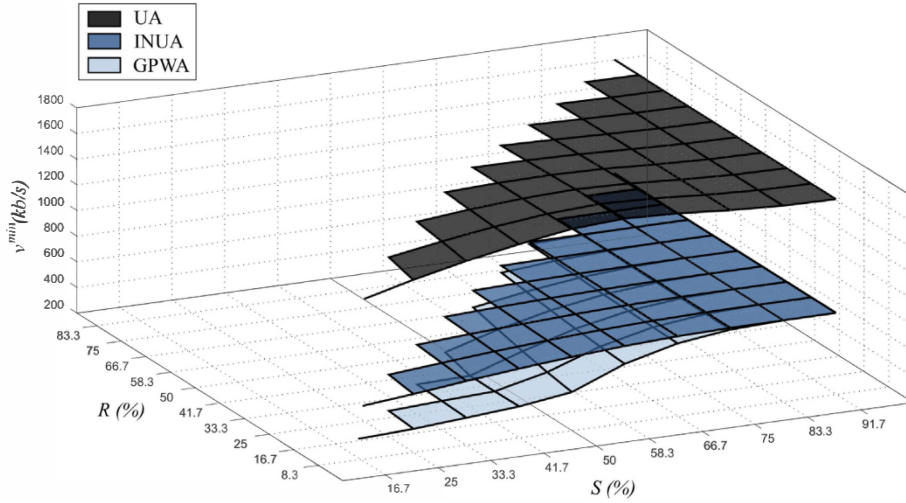


Fig. 8. Minimum upload rate required from servers to guarantee abundance conditions as a function of  $S$  and  $R$ ; for  $N = 24$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.6$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.5c$ .

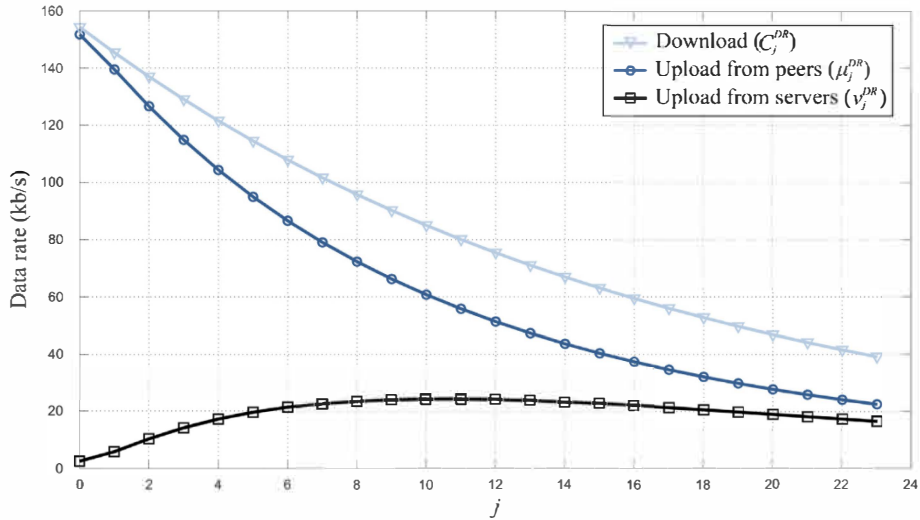


Fig. 9. Download and upload data rates for each downloading group ( $j$ ), considering GPWA,  $N = 24$ ,  $S = 22$ ,  $R = 20$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.5c$ .

can share content). In addition, for small values of  $S$  the demand of very low windows is not reduced, as a result, in INUA these windows maintain a high dependence on servers to achieve abundance, whereas GPWA is directly benefitted from downloading restrictions. From the previous comparisons, we conclude that GPWA efficiently operates in a wider variety of scenarios than recent previous proposals.

Additionally, by contrasting the evaluation of GPWA for  $\gamma = 0.006$  and  $\gamma = 0.6$  (Figs. 7 and 8, respectively), we observe that this scheme is fairly insensitive to the level of cooperation of the seeders, because it efficiently distributes resources from leechers and regular seeders which do not require incentives, other than their own download and/or playback processes, to stay in the system.

In Fig. 9, we depict the data rate  $C_j$ . We also show how much of these rates is supplied by the peers ( $\mu_j$ ) and how much by the CDN ( $v_j$ ). As expected,  $C_j$  is monotonically decreasing, as a result of peers abandoning the system. Also notice that low windows are almost exclusively supplied by the P2P network, however, even for high windows, this network is the main supplier. This comes as a result of considering the P2P network as the primary source and efficiently using its resources.

In Fig. 10, we present the download rate  $C$  as well as the upload rates from each source:  $\mu$  (from the P2P network) and  $v$  (from the CDN). As

expected, small amounts of prefetched windows (small values of  $S$  and  $R$ ) decrease the download data rate in the network, and this reduction primarily benefit the CDN.

Despite the advantages of reducing  $S$ , this comes with a QoS degradation, as we illustrate in Fig. 11, where the proportion of stalled peers ( $\alpha_{st}$ ) is depicted. We observe that  $\alpha_{st}$  is degraded as  $S$  and  $R$  decrease; hence, the values for  $S$  and  $R$  should be selected as a trade-off between  $\alpha_{st}$  and  $C$ .

From Fig. 11 we also observe that the *playback rate to download rate* ratio ( $p/c$ ) has a significant effect on  $\alpha_{st}$ . This is expected, since if  $p$  approximates  $c$  the probability of suffering a stalling event increases.

## 10.2. Results for the IF scenario

According to measurements reported in Chen et al. (2014), the playback positions after a random-peek are not necessarily uniformly distributed along the video length; however, for the sake of simplicity, in this section, we assume that  $q_{k,l} = q$  holds, for all  $k, l$ . We select  $q$  according to the average random-peek rate reported in the same reference. The authors identified that in short videos there are approximately 4.76 random-peeks per playback which is equivalent to

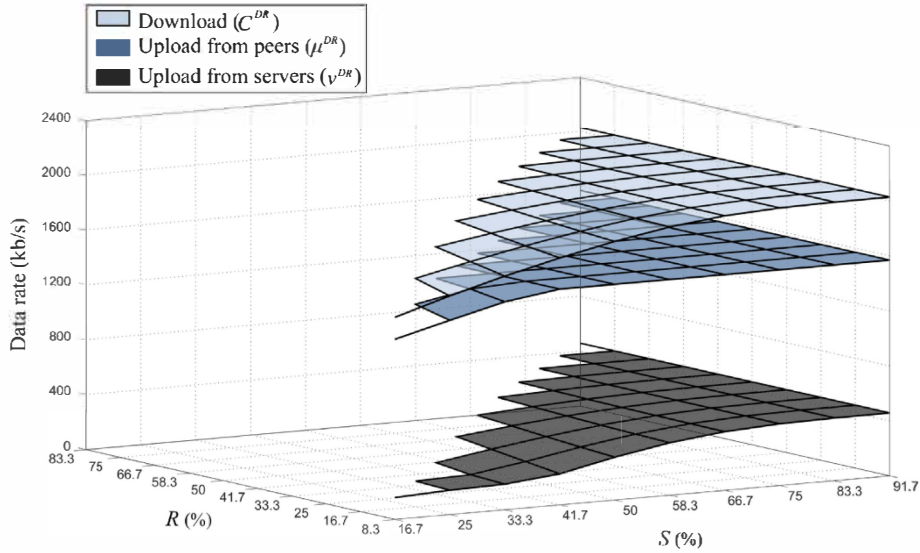


Fig. 10. Download and upload data rates in terms of  $S$  and  $R$ , considering GPWA,  $N = 24$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.5c$ .

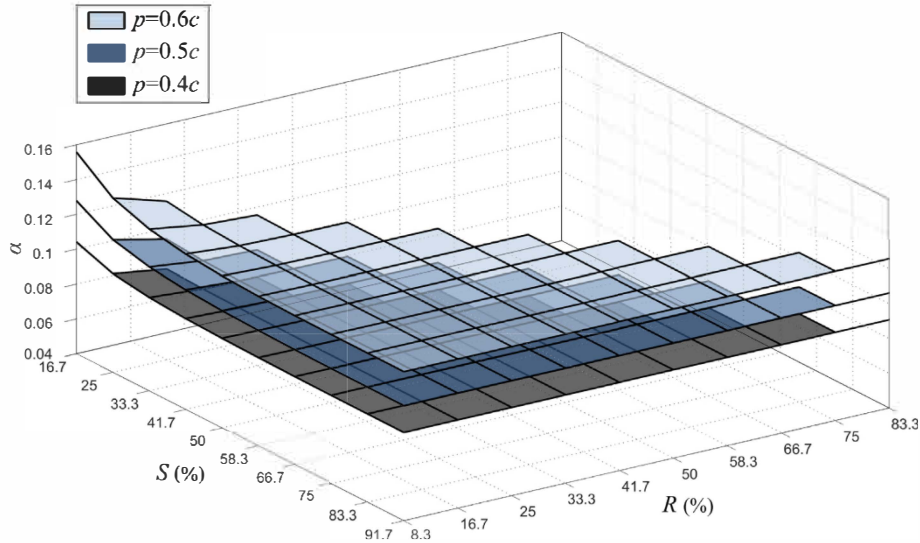


Fig. 11. Proportion of stalled leechers in terms of  $S$ ,  $R$  and  $p$ , considering GPWA,  $N = 24$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$  and  $\mu = 0.002544$ .

a random-seek rate of  $q = 0.00097$  in our framework. Considering this, we evaluate our system for  $0 \leq q \leq 0.001$ .

Fig. 12 shows the behavior of the number of peers in the system as a function of  $q$ . As expected the number of peers in plane 0 ( $X^0$ ) decreases as  $q$  increases, however, in spite of the large number of planes ( $N = 24$ ), there are high concentrations of peers in this plane. This is true even for the largest evaluated value of  $q$ , because peers that do not experience random-seeks tend to stay in the system for longer periods of time (while playing the video). However, resources from the P2P network are affected by random-seeks, as evidenced by the decreasing number of seeders in plane 0 ( $X_N^0$ ) in terms of  $q$  (recall that these are the only peers in possession of the whole video).

In Fig. 12 we can also appreciate that the fluid model defined in (65)-(72) and the Markov chain described in Table 2 converge to the same solution.

On the other hand, in Fig. 13, we present the data rates demanded by each window, for  $q = 0.008$ . It is interesting to notice that, in this scenario, the upload data rate provided by servers is very small (or even zero) for the highest windows (e.g.  $21 \leq j \leq 23$ ). This is a result of

the existence of partial seeders, which can only serve peers in those windows.

As we show in Fig. 14, moderate increases of  $q$  have little effect on the upload from servers. This is a consequence, again, of the existence of partial seeders, whose contributions *substitute* the reduction of resources from regular seeders, as we observe in Fig. 12. Notice that this substitution is possible because we are considering a scheme (GPWA) that is capable of distributing resources in terms of the downloading progress.

Lastly, in Fig. 15, we show  $\alpha_{st}$  as a function of  $q$ . As expected, this performance parameter is largely affected by increasing the rate of random-seeks. However, we must highlight that it also considers the unavoidable playback interruptions after random-seeks outside buffered windows.

## 11. Conclusions and future work

In this paper, we first review, in Section 3, a basic window-based model that abstracts the dynamics of a population of peers playing,

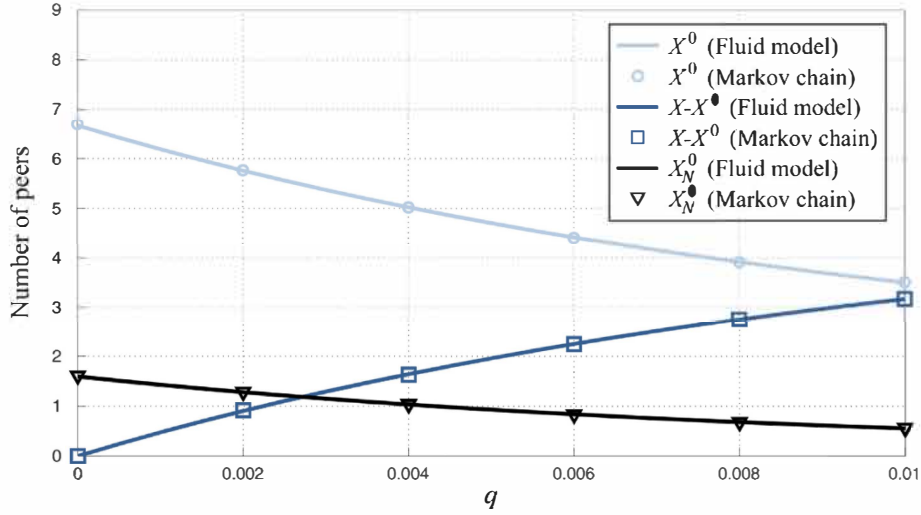


Fig. 12. Number of peers in the system, in terms of  $q$ , considering  $N = 24$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.5c$ .

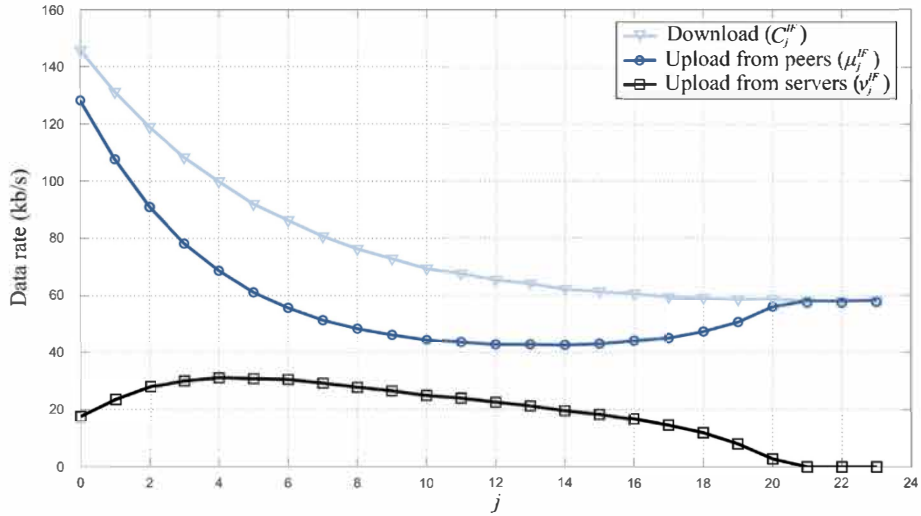


Fig. 13. Download and upload data rates for each downloading group ( $j$ ), considering GPWA,  $N = 24$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$ ,  $\mu = 0.002544$ ,  $p = 0.5c$  and  $q = 0.008$ .

downloading and sharing a video file of common interest. This model provides us with expressions to estimate the number of peers in each window, (16)-(22), as well as the abundance conditions when the UA scheme is implemented, (29). In Section 4, we extend our model to the case where resources are assigned according to the GPWA scheme (which avoids that leechers in high windows starve due to the lack of resources), and we provide expressions to calculate the amount of resources from servers that guarantee abundance in the system, (34). In Section 6, we further develop the basic model to consider the practical case where peers can only download and maintain in the buffers a certain number of chunks; in this case, we also obtain expressions for the number of peers per group, (46)-(57), and the amount of resources from servers to achieve abundance, (61); moreover, similar expressions are obtained in Section 7 for the scenario where interaction functionalities (pause, forward and backward seek) are enabled, (65)-(72) and (77).

Through this analysis, we showed that the proposed analytical framework is expressive enough to accurately describe the performance of VoD services over hybrid CDN-P2P systems under a wide range of scenarios, including different resource allocation schemes, and different assumptions about end-users behavior as well as about restrictions of the end devices.

In addition, we show that the closed-form expressions derived from our models allowed us to characterize the conditions needed to guarantee an abundance in the whole system. This is particularly important because abundance conditions are needed to guarantee that the system will achieve targets of QoS.

The numerical results derived from the analytical models also revealed a number of relevant insights about the behavior of this kind of systems. Among them, we would like to highlight the following:

- The prefetching strategies in P2P networks must be a trade-off between the efficient utilization of network resources and the satisfaction of QoS parameters (e.g. start-up delay and stalling events).
- The amount of resources from servers can be effectively reduced by distribution schemes based on the current progress of the downloading processes. This is true even for moderate random-seek rates.

As future work, we plan to tackle a number of problems related to finding optimal operation parameters of the system. In particular, we are interested in determining resource allocation schemes that minimize the amount of resources from servers or from peers. We also plan to extend this framework to incorporate details about mobile networks with particular focus on minimizing upload-bandwidth cost and energy consumption.



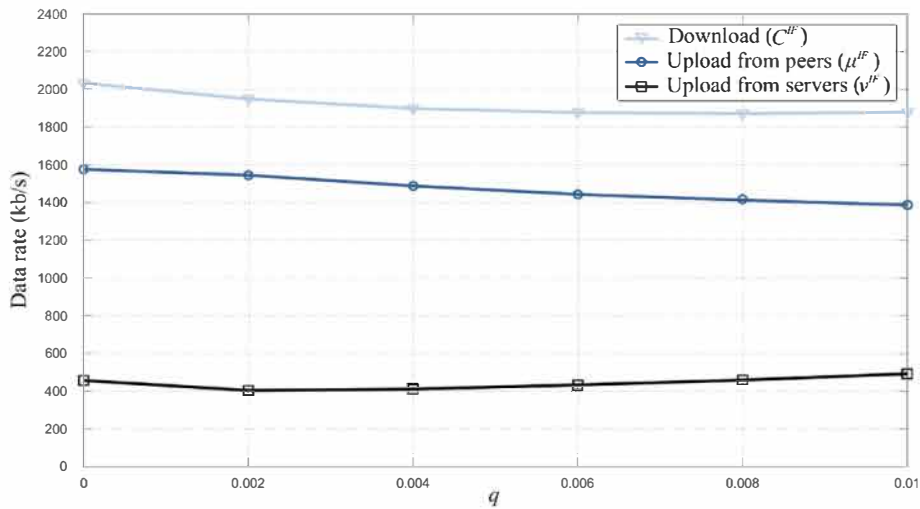


Fig. 14. Download and upload data rates in terms of  $q$ , considering GPWA,  $N = 24$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.5c$ .

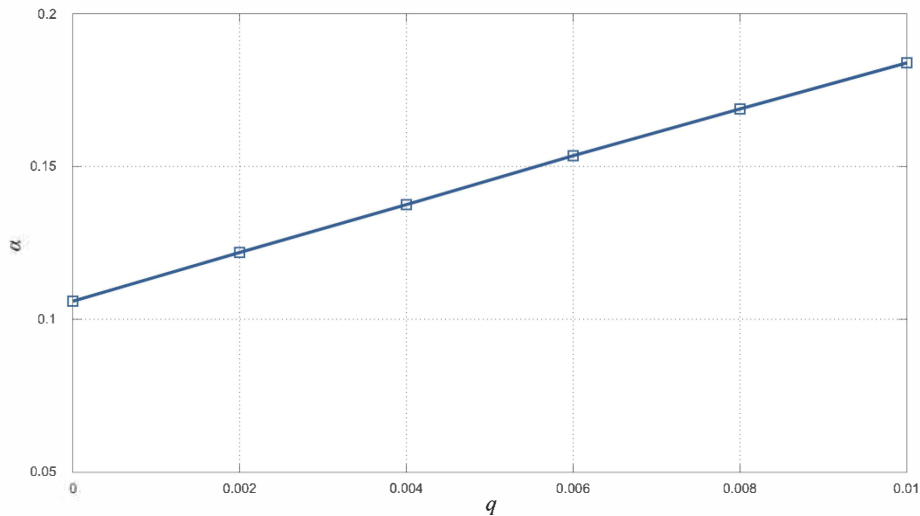


Fig. 15. Proportion of stalled leechers in terms of  $q$ , considering GPWA,  $N = 24$ ,  $\lambda = 0.04$ ,  $\theta = 0.006$ ,  $\gamma = 0.006$ ,  $c = 0.00407$ ,  $\mu = 0.002544$  and  $p = 0.5c$ .

Lastly, we identified some limitations of our analysis that also can be considered future work:

- We do not consider variable video-coding rates, which is a basic component of the DASH scheme. This can be solved by permitting variations in the values of the downloading and uploading rates ( $c$  and  $\mu$ ) at the beginning of every window.
- We consider the existence of a unique swarm of peers interested in the same content; however, it is possible to model swarms interaction by defining the rates at which peers transit from one swarm to another (peers that switch from one video to another) and the upload rate that one swarm can provide to another (e.g. when peers download and share content that they are not playing).
- Even though, this paper is focused on VoD, other services, e.g. live video, can also be analyzed with our framework. This can be achieved by modifying the set of windows that can be shared among peers.

#### CRedit authorship contribution statement

**Noé Torres-Cruz:** Conceptualization, Methodology, Software, Validation, Writing - original draft, Visualization, Project administration. **Mario E. Rivero-Angeles:** Conceptualization, Methodology, Writing - review & editing, Supervision. **Gerardo Rubino:** Conceptualiza-

tion, Methodology, Writing - review & editing, Supervision. **Ricardo Menchaca-Mendez:** Formal analysis. **Rolando Menchaca-Mendez:** Formal analysis, Writing - review & editing. **David Ramirez:** Writing - review & editing.

#### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### References

- Aspirot, L., Mordecki, E., Rubino, G., Sept. 2011. Fluid limits applied to peer to peer network analysis. In: Proc. 8th International Conference on Quantitative Evaluation of Systems (QEST), pp. 13–20.
- Baez-Esquivel, E., Rivero-Angeles, M.E., Rubino, G., Jun. 2013. Priority scheme for window-based video-on-demand transmission on BitTorrent-like peer-to-peer networks. In: Proc. IEEE International Conference on Communications (ICC), pp. 3000–3005.
- Bethanabhotla, D., Caire, G., Neely, M.J., Jan. 2015. Adaptive video streaming for wireless networks with multiple users and helpers. *IEEE Trans. Commun.* 63 (1), 268–285.
- Chang, L., Pan, J., Xing, M., Sept. 2013. Effective utilization of user resources in PA-VoD systems with channel heterogeneity. *IEEE J. Sel. Area. Commun.* 31 (9), 227–236.
- Chen, L., Zhou, Y., Chiu, D.M., June 2014. A study of user behavior in online VoD services. *Comput. Commun.* 46 (23), 66–75.

- Choi, J., Reaz, A., Mukherjee, B., 2012. A survey of user behavior in VoD service and bandwidth-saving multicast streaming schemes. *IEEE Commun. Surv. Tutor.* 14 (1), 156–169.
- Ciullo, et al., July 2014. Peer-assisted VoD systems: an efficient modeling framework. *IEEE Trans. Parallel Distr. Syst.* 25 (7), 1852–1863.
- Cong, X., Shuang, K., May 2015. SBDP: bandwidth prediction mechanism towards server demands in P2P-VoD system. *Peer-to-Peer Netw. Appl.* 8 (3), 501–511.
- De Veciana, G., Yang, X., 2003. Fairness, incentives and performance in peer-to-peer networks. In: *Proc. of the IEEE Forty-First Annual Allerton Conference on Control, Communications and Computing.*
- Deltouzos, K., Denazis, S., July 2015. Distributed energy-efficient peer-to-peer VoD system. In: *Proc. of the IEEE Symposium on Computers and Communication (ISCC)*, pp. 863–868.
- Dimopoulos, G., Barlet-Ros, P., Sanjuas-Cuxart, J., Oct. 2013. Analysis of YouTube user experience from passive measurements. In: *Proc. IEEE 9th International Conference on Network and Service Management (CNSM)*, pp. 260–267.
- Dubin, R., et al., 2015. Hybrid clustered peer-assisted DASH-SVC system. In: *Proc. IEEE International Conference on CIT/UCC/DAS/PICOM*, pp. 1651–1656.
- Ericsson Mobility Report, November 2018. Internet: <https://www.ericsson.com/en/mobility-report/reports> March 19, 2019.
- Ethier, S.N., Kurtz, T., 1986. *Markov Processes: Characterization and Convergence.* J. Wiley & Sons, New York, NY.
- Faiqurahman, M., Kistijantoro, A.I., May 2015. Implementation of modified Probabilistic Caching Schema on Bittorrent protocol for video on demand content. In: *Proc. International Seminar on Intelligent Technology and its Applications (ISITIA)*, pp. 357–362.
- Gramatikov, S., Jaureguizar, F., April 2016. Modelling and analysis of non-cooperative peer-assisted VoD streaming in managed networks. *Multimed. Tool. Appl.* 75 (8), 4321–4348.
- Haghighi, A.A., Heydari, S.S., Shahbazpanah, S., 2018. Dynamic QoS-aware resource assignment in cloud-based content-delivery networks. *IEEE Access* 6, 2298–2309.
- Hossfeld, T., et al., Jul. 2012. Initial delay vs. interruptions: between the devil and the deep blue sea. In: *Proc. IEEE Fourth International Workshop on Quality of Multimedia Experience (QoMEX)*, pp. 1–6.
- Huang, G., et al., 2014. An upload bandwidth allocation algorithm in data scheduling of P2P VoD system. In: *Proc. IEEE 5th International Conference on Software Engineering and Service Science*, pp. 435–438.
- Hwang, K., et al., Sept. 2016. Joint-family: adaptive bitrate video-on-demand streaming over peer-to-peer networks with realistic abandonment patterns. *Comput. Network.* 106 (4), 226–244.
- Jia, S., et al., 2016. Modeling of P2P-based video sharing performance for content-oriented. [Online] *Mobile Inf. Syst.*, <https://doi.org/10.1155/2016/1319497>. Article ID 1319497. Available: (Accessed 12 January 2017).
- Li, Z., Wu, Q., Salamatian, K., Xie, G., June 2015. Video Delivery performance of a large-scale VoD system and the implications on content Delivery. *IEEE Trans. Multimed.* 17 (6), 880–892.
- Qiu, D., Srikant, R., Oct. 2004. Modeling and performance analysis of BitTorrent-like peer-to-peer networks. In: *Proc. Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.*
- Ramos-Muoz, J.J., et al., Feb. 2014. Characteristics of mobile Youtube traffic. *IEEE Wirel. Commun.* 21 (1), 18–25.
- Rivero-Angeles, M.E., Rubino, G., May 2010. Priority-based scheme for file distribution in peer-to-peer networks. In: *Proc. IEEE International Conference on Communications (ICC)*, pp. 1–6.
- Rohmer, T., Nakib, A., Nafaa, A., Jan.-Feb. 2015. A learning-based resource allocation approach for P2P streaming systems. *IEEE Netw.* 29 (1), 4–11.
- Romero, P., Robledo, F., Rodriguez-Boca, P., Rostagnol, C., March 2015. Lyapunov stability and performance of user-assisted Video-on-Demand services. *Comput. Network.* 79 (14), 203–205.
- Saengarunwong, A., Sanguankotchakorn, T., Oct. 2018. A two-step server selection in hybrid CDN-P2P mesh-based for video-on-demand streaming. In: *Proc. of International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 400–504.
- Sasabe, M., July 2018. Analysis of optimal piece flow in tit-for-tat-based P2P streaming. *Comput. Network.* 139 (5), 60–69.
- Sheshjavani, A.G., Akbari, B., March 2017. An adaptive buffer-map exchange mechanism for pull-based peer-to-peer video-on-demand streaming systems. *Multimed. Tool. Appl.* 76 (5) pp. 75357575-61.
- The Global Internet Phenomena Report, 2019. Internet: <https://www.sandvine.com/phenomena> March 19, 2019.
- Tian, C., Suna, J., Wua, W., Luo, Y., Nov. 2015. Optimal bandwidth allocation for hybrid Video-on-Demand streaming with a distributed max flow algorithm. *Comput. Network.* 91 (14), 483–494.
- Torres-Cruz, N., et al., 2017. A window-based, server-assisted P2P network for VoD services with QoE guarantees. *Mobile Inf. Syst.* 18, <https://doi.org/10.1155/2017/2084684>. 2017 Article ID 2084684.
- Torres-Cruz, N., et al., Dec. 2018. An efficient resource allocation scheme for VoD services over window-based P2P networks. *Multimed. Tool. Appl.* 77 (23), 31427–31445.
- Wichtlhuber, M., et al., Sept. 2015. QTrade: a quality of experience based peercasting trading scheme. In: *Proc. IEEE International Conference on Peer-To-Peer Computing (P2P)*, pp. 1–10.
- Wu, W., Ma, R.T.B., Lui, J.C.S., March 2014. Distributed caching via rewarding: an incentive scheme design in P2P-VoD systems. *IEEE Trans. Parallel Distr. Syst.* 25 (3), 612–621.
- Zhang, G., Liu, W., Hei, X., Cheng, W., Feb. 2015. Unreeling Xunlei Kankan: understanding hybrid CDN-P2P video-on-demand streaming. *IEEE Trans. Multimed.* 17 (3), 229–242.
- Zhao, Z.W., Samarth, S., Ooi, W.T., May 2013. Modeling the effect of users interactions on mesh-based P2P VoD streaming system. *ACM Trans. Multimed. Comput. Commun. Appl.* 9 (No. 2).
- Zhao, J., Wu, C., Lin, X., March 2015. Locality-aware streaming in hybrid P2P-cloud CDN systems. *Peer-to-Peer Netw. Appl.* 8 (2), 320–335.
- Zhao, C., Zhao, J., Lin, X., Wu, C., Oct. 2016. Capacity of P2P on-demand streaming with simple, robust, and decentralized control. *IEEE/ACM Trans. Netw.* 24 (5), 2607–2620.
- Zhou, Y., Fu, T.Z.J., Chiu, D.M., Aug. 2015. A unifying model and analysis of P2P VoD replication and scheduling. *IEEE/ACM Trans. Netw.* 23 (4), 1163–1175.