Performance Analysis of Non-stationary Peer-assisted VoD Systems

Delia Ciullo[1], Valentina Martina[1], Michele Garetto[2], Emilio Leonardi[1], Gianluca Torrisi[3]

[1]Politecnico di Torino

[2]Università di Torino

[3]CNR - Instituto per le Applicazioni di Calcolo

March 26-th, 2012

Introduction

In **peer-assisted Video-on-Demand (VoD) systems:**

Introduction

In **peer-assisted Video-on-Demand (VoD) systems:**

- users browse a catalog of available videos and asynchronously issue requests to watch a given content;

Introduction

In **peer-assisted Video-on-Demand (VoD) systems:**

- users browse a catalog of available videos and asynchronously issue requests to watch a given content;
- content is divided into chunks that can be retrieved either from

Introduction

In **peer-assisted Video-on-Demand (VoD) systems:**

- users browse a catalog of available videos and asynchronously issue requests to watch a given content;
- content is divided into chunks that can be retrieved either from
  - the servers

Introduction

In **peer-assisted Video-on-Demand (VoD) systems:**

- users browse a catalog of available videos and asynchronously issue requests to watch a given content;
- content is divided into chunks that can be retrieved either from
  - the servers
  - other peers currently retrieving the same video (leechers)

Introduction

In **peer-assisted Video-on-Demand (VoD) systems:**

- users browse a catalog of available videos and asynchronously issue requests to watch a given content;
- content is divided into chunks that can be retrieved either from
    - the servers
    - other peers currently retrieving the same video (leechers)
    - peers storing the whole video (seeds);

Introduction

In **peer-assisted Video-on-Demand (VoD) systems:**

- users browse a catalog of available videos and asynchronously issue requests to watch a given content;
- content is divided into chunks that can be retrieved either from
  - the servers
  - other peers currently retrieving the same video (leechers)
  - peers storing the whole video (seeds);
- chunks must be retrieved by peers almost in sequence to guarantee small play-out delays;

Introduction

In **peer-assisted Video-on-Demand (VoD) systems:**

- users browse a catalog of available videos and asynchronously issue requests to watch a given content;
- content is divided into chunks that can be retrieved either from
  - the servers
  - other peers currently retrieving the same video (leechers)
  - peers storing the whole video (seeds);
- chunks must be retrieved by peers almost in sequence to guarantee small play-out delays;
- a minimum average download rate equal to the video playback rate must be sustained to guarantee service continuity; the system (exploiting servers bandwidth when needed) is able to steadily meet this constraint.

Assumptions

Assumptions

- Video is downloaded by each user at constant rate d, greater or equal to the playback rate d_v;

Assumptions

- Video is downloaded by each user at constant rate d, greater or equal to the playback rate d_v;
- upload available bandwidth U_i of peer i is a random variable with a assigned distribution (U_i are i.i.d.);

Assumptions

- Video is downloaded by each user at constant rate d, greater or equal to the playback rate d_v;
- upload available bandwidth U_i of peer i is a random variable with a assigned distribution (U_i are i.i.d.);
- users contribute their upload bandwidth to the video distribution as long as they are in the system;

Assumptions

- Video is downloaded by each user at constant rate d, greater or equal to the playback rate d_v;
- upload available bandwidth U_i of peer i is a random variable with a assigned distribution (U_i are i.i.d.);
- users contribute their upload bandwidth to the video distribution as long as they are in the system;
- the arrival process of requests (and users) for a video is a (possibly non-homogeneous) Poisson process with intensity $\lambda(t)$;

Assumptions

- Video is downloaded by each user at constant rate d, greater or equal to the playback rate d_v;
- upload available bandwidth U_i of peer i is a random variable with a assigned distribution (U_i are i.i.d.);
- users contribute their upload bandwidth to the video distribution as long as they are in the system;
- the arrival process of requests (and users) for a video is a (possibly non-homogeneous) Poisson process with intensity $\lambda(t)$;
- user's sojourn time is described by an arbitrary random variable T with finite mean $\overline{T}$ and complementary cumulative distribution function $G_T(x)$.

Preliminary Observations

- The number of active users $N(t)$ follows a Poisson distribution with mean

$$\overline{N}(t) = \int_0^\infty \lambda(t-x) G_T(x) \, \mathrm{d}x;$$

Preliminary Observations

- The number of active users $N(t)$ follows a Poisson distribution with mean
$$\overline{N}(t) = \int_0^\infty \lambda(t-x) G_T(x)\,\mathrm{d}x;$$
- $\tau_d = L/d$ is the time needed to download the whole video, and $\overline{T}_d = \int_0^{\tau_d} G_T(x)dx$ is the average time spent by peers downloading the video, taking into account premature abandonments;

Preliminary Observations

- The number of active users $N(t)$ follows a Poisson distribution with mean
$$\overline{N}(t) = \int_0^\infty \lambda(t-x) G_T(x)\,\mathrm{d}x;$$
- $\tau_d = L/d$ is the time needed to download the whole video, and $\overline{T}_d = \int_0^{\tau_d} G_T(x)dx$ is the average time spent by peers downloading the video, taking into account premature abandonments;
- $N_d(t)$ is the number of downloading users with mean $\overline{N}_d(t) = \int_0^{\tau_d} \lambda(t-x) G_T(x)\,\mathrm{d}x$, and $N_{\text{seed}}(t)$ the number of seeds with mean $\overline{N}_{\text{seed}}(t) = \overline{N}(t) - \overline{N}_d(t)$;

Preliminary Observations

- The number of active users $N(t)$ follows a Poisson distribution with mean

$$\overline{N}(t) = \int_0^{\infty} \lambda(t-x) G_T(x)\, \mathrm{d}x;$$

- $\tau_d = L/d$ is the time needed to download the whole video, and $\overline{T}_d = \int_0^{\tau_d} G_T(x) dx$ is the average time spent by peers downloading the video, taking into account premature abandonments;
- $N_d(t)$ is the number of downloading users with mean $\overline{N}_d(t) = \int_0^{\tau_d} \lambda(t-x) G_T(x)\, \mathrm{d}x$, and $N_{\mathrm{seed}}(t)$ the number of seeds with mean $\overline{N}_{\mathrm{seed}}(t) = \overline{N}(t) - \overline{N}_d(t)$;
- we define the average system load as:

$$\gamma = \frac{d\overline{T}_d}{\overline{U}\ \overline{T}}.$$

Goal

Our goal is:

[1] D.Ciullo, V.Martina, M.Garetto, E.Leonardi, G.L.Torrisi, “Stochastic Analysis of Self-Sustainability in Peer-Assisted VoD Systems”, Session TS05, Thursday, h. 8.30-10.00

Goal

Our goal is:

- to characterize the bandwidth requested from the servers S (and its average $\overline{S}$);

[1] D.Ciullo, V.Martina, M.Garetto, E.Leonardi, G.L.Torrisi, "Stochastic Analysis of Self-Sustainability in Peer-Assisted VoD Systems", Session TS05, Thursday, h. 8.30-10.00

Goal

Our goal is:

- to characterize the bandwidth requested from the servers S (and its average $\overline{S}$);
  - we develop an approximate efficient and accurate fluid model to compute S;

[1] D.Ciullo, V.Martina, M.Garetto, E.Leonardi, G.L.Torrisi, "Stochastic Analysis of Self-Sustainability in Peer-Assisted VoD Systems", Session TS05, Thursday, h. 8.30-10.00

Goal

Our goal is:

- to characterize the bandwidth requested from the servers S (and its average $\overline{S}$);
  - we develop an approximate efficient and accurate fluid model to compute S;
  - our approach is able to capture several stochastic effects related to peer churn, upload bandwidth heterogeneity, non-stationary traffic conditions;

---

[1] D.Ciullo, V.Martina, M.Garetto, E.Leonardi, G.L.Torrisi, "Stochastic Analysis of Self-Sustainability in Peer-Assisted VoD Systems", Session TS05, Thursday, h. 8.30-10.00

Goal

Our goal is:

- to characterize the bandwidth requested from the servers S (and its average $\overline{S}$);
  - we develop an approximate efficient and accurate fluid model to compute S;
  - our approach is able to capture several stochastic effects related to peer churn, upload bandwidth heterogeneity, non-stationary traffic conditions;
  - our methodology can be exploited to design efficient peer-assisted VoD systems and optimal resource allocation strategies.

---

[1] D.Ciullo, V.Martina, M.Garetto, E.Leonardi, G.L.Torrisi, "Stochastic Analysis of Self-Sustainability in Peer-Assisted VoD Systems", Session TS05, Thursday, h. 8.30-10.00

Goal

Our goal is:

- to characterize the bandwidth requested from the servers S (and its average $\overline{S}$);
  - we develop an approximate efficient and accurate fluid model to compute S;
  - our approach is able to capture several stochastic effects related to peer churn, upload bandwidth heterogeneity, non-stationary traffic conditions;
  - our methodology can be exploited to design efficient peer-assisted VoD systems and optimal resource allocation strategies.

In [1] we obtain rigorous bounds for the sequential delivery scheme and asymptotic results as the number of users increases.

---

[1] D.Ciullo, V.Martina, M.Garetto, E.Leonardi, G.L.Torrisi, "Stochastic Analysis of Self-Sustainability in Peer-Assisted VoD Systems", Session TS05, Thursday, h. 8.30-10.00

A simple Lower Bound

A simple universal lower bound to $\overline{S}(t)$ for any chunk distribution scheme is

$$\overline{S}(t) \leq \max\{0, d\overline{N}_d(t) - \overline{U}\ \overline{N}(t)\}.$$

Intuition: The additional server bandwidth is given by users requested

bandwidth minus their total upload bandwidth.

Note that this trivial lower bound was already shown in: C. Huang, J. Li, and K. W. Ross, Can Internet Video-on-Demand Be Profitable? in ACM SIGCOMM, 2007.

Analysis

Analysis

Let S_d be the aggregate bandwidth requested by the downloading users. The aggregate upload bandwidth offered by the seeds is

$$S_{\text{seed}} = \sum_{i=1}^{N_{\text{seed}}} U_i.$$

Analysis

Let S_d be the aggregate bandwidth requested by the downloading users. The aggregate upload bandwidth offered by the seeds is

$$S_{\text{seed}} = \sum_{i=1}^{N_{\text{seed}}} U_i.$$

The bandwidth requested from the servers is:

$$S \triangleq \max\{0, S_d - S_{\text{seed}}\}$$

where S_d is the bandwidth demanded by downloading peers.

Analysis(2)

We define $S_d(k) \triangleq (S_d(t) \mid N_d(t) = k)$

Analysis(2)

We define $S_d(k) \triangleq (S_d(t) \mid N_d(t) = k)$

Theorem

$S_d(k)$ satisfies the following recursive equation:

$$S_d(k) = \begin{cases} d & k = 1 \\ d + \max\{0, S_d(k-1) - U_k\} & k > 1 \end{cases}$$

Analysis(3)

Analysis(3)

- We characterize the distribution of the server bandwidth using a second-order approximation;

Analysis(3)

- We characterize the distribution of the server bandwidth using a second-order approximation;
- we approximate the distribution of the quantity $S_d(k-1) - U_k$ (for each $k \geq 2$) with a normal distribution matching the first two moments of this quantity;

Analysis(3)

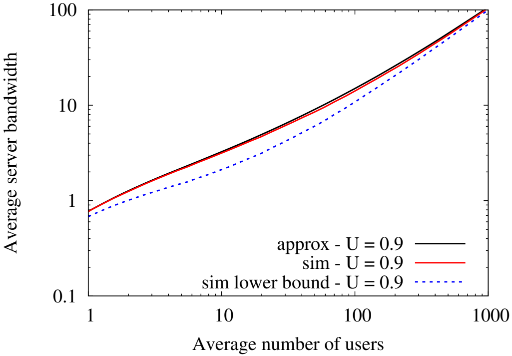
- We characterize the distribution of the server bandwidth using a second-order approximation;
- we approximate the distribution of the quantity $S_d(k-1) - U_k$ (for each $k \geq 2$) with a normal distribution matching the first two moments of this quantity;
- we apply standard formulas of the truncated normal distribution to derive the first two moments of $S_d(k)$ as a function of the first two moments of $S_d(k-1)$;

Analysis(3)

- We characterize the distribution of the server bandwidth using a second-order approximation;
- we approximate the distribution of the quantity $S_d(k-1) - U_k$ (for each $k \geq 2$) with a normal distribution matching the first two moments of this quantity;
- we apply standard formulas of the truncated normal distribution to derive the first two moments of $S_d(k)$ as a function of the first two moments of $S_d(k-1)$;
- a similar approximation is subsequently applied to take into account the effect of the seeds.

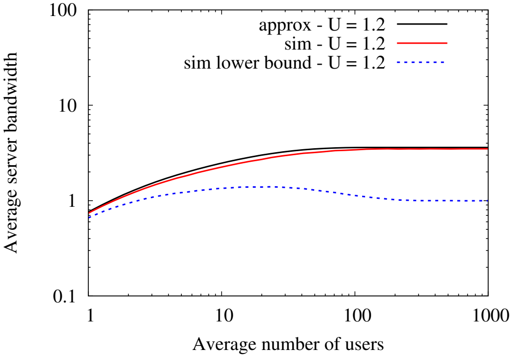
Swarm size effect

$$d = d_v = 1,\ \overline{T} = \overline{T}_d = \tau_d$$



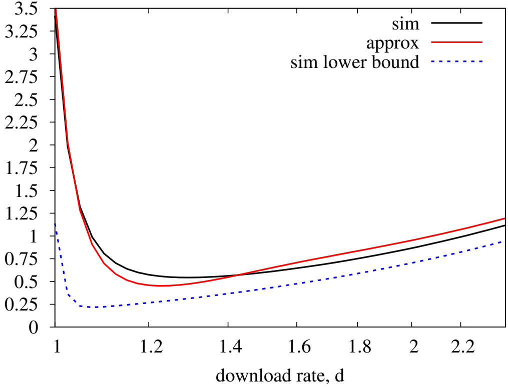
Swarm size effect

$$d = d_v = 1,\ \overline{T} = \overline{T}_d = \tau_d$$



Download rate impact

$$U = 1.2,\ d_v = 1,\ \overline{T} = \overline{T}_d = \tau_d$$



Extension to non sequential download

Extension to non sequential download

- A common approach in P2P-VoD is to allow users to receive also out-of-sequence chunks of the video within a limited sliding window of data starting from the point currently played;

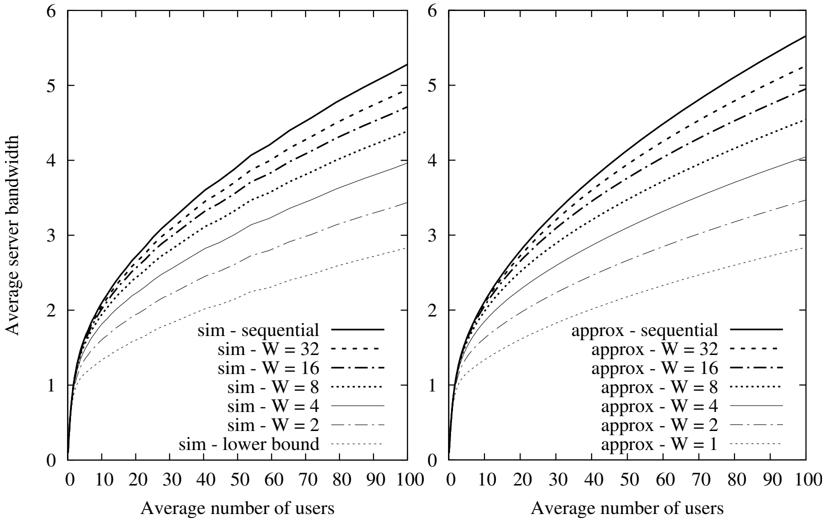
Extension to non sequential download

- A common approach in P2P-VoD is to allow users to receive also out-of-sequence chunks of the video within a limited sliding window of data starting from the point currently played;
- for simplicity, instead of considering an actual sliding window, we divide the video into a fixed number W of non-overlapping segments of size $L_W = L/W$;

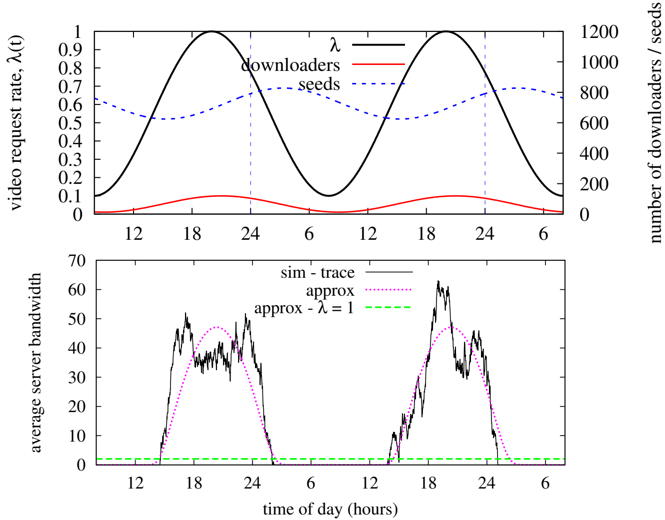
Extension to non sequential download

- A common approach in P2P-VoD is to allow users to receive also out-of-sequence chunks of the video within a limited sliding window of data starting from the point currently played;
- for simplicity, instead of considering an actual sliding window, we divide the video into a fixed number W of non-overlapping segments of size $L_W = L/W$;
- we assume that, within a segment, chunk based out-of-sequence distribution can be exploited, and we extend our approximate model to deal with partially non-sequential chunk delivery.

Impact of non-sequential delivery



Impact of non stationarity



Conclusions

- We have proposed a computationally-efficient methodology to estimate the server bandwidth requested in non-stationary P2P-VoD systems;

Conclusions

- We have proposed a computationally-efficient methodology to estimate the server bandwidth requested in non-stationary P2P-VoD systems;
- we have discovered several interesting properties:
  - the server bandwidth can be minimized by a proper selection of the download rate;

Conclusions

- We have proposed a computationally-efficient methodology to estimate the server bandwidth requested in non-stationary P2P-VoD systems;
- we have discovered several interesting properties:
  - the server bandwidth can be minimized by a proper selection of the download rate;
  - the server bandwidth increases with the variation coefficient of the peer upload bandwidth;

Conclusions

- We have proposed a computationally-efficient methodology to estimate the server bandwidth requested in non-stationary P2P-VoD systems;
- we have discovered several interesting properties:
  - the server bandwidth can be minimized by a proper selection of the download rate;
  - the server bandwidth increases with the variation coefficient of the peer upload bandwidth;
  - the gain achievable by non-sequential schemes over the simple sequential scheme depends critically on the size of the sliding window and the number of downloading users;

Conclusions

- We have proposed a computationally-efficient methodology to estimate the server bandwidth requested in non-stationary P2P-VoD systems;
- we have discovered several interesting properties:
  - the server bandwidth can be minimized by a proper selection of the download rate;
  - the server bandwidth increases with the variation coefficient of the peer upload bandwidth;
  - the gain achievable by non-sequential schemes over the simple sequential scheme depends critically on the size of the sliding window and the number of downloading users;
  - non-stationary systems are affected by a misalignment problem between downloaders and seeds.

Thank you!