Second Generation P2P Live Streaming

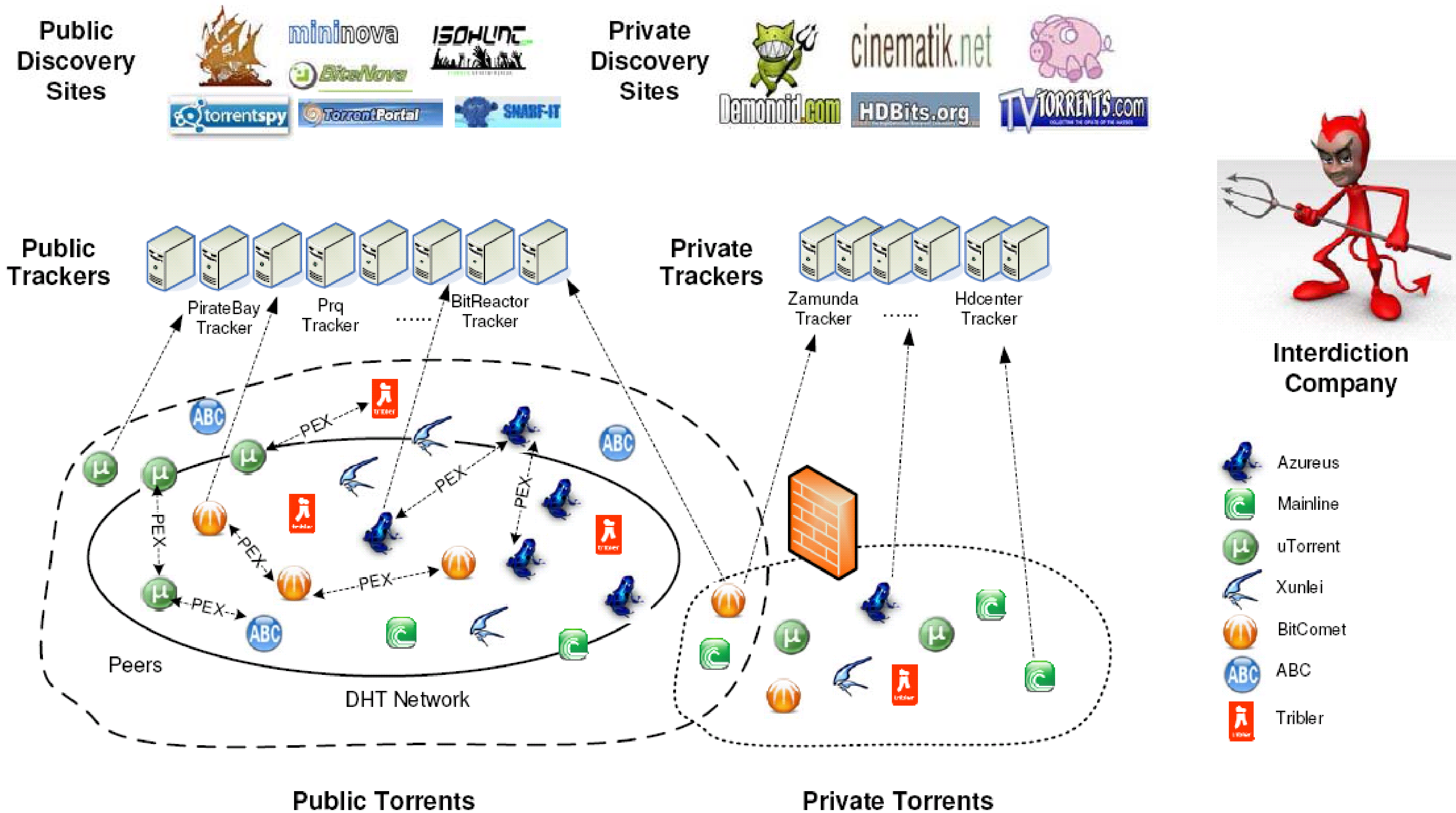
Keith Ross

Polytechnic Institute of NYU

Some P2P success stories

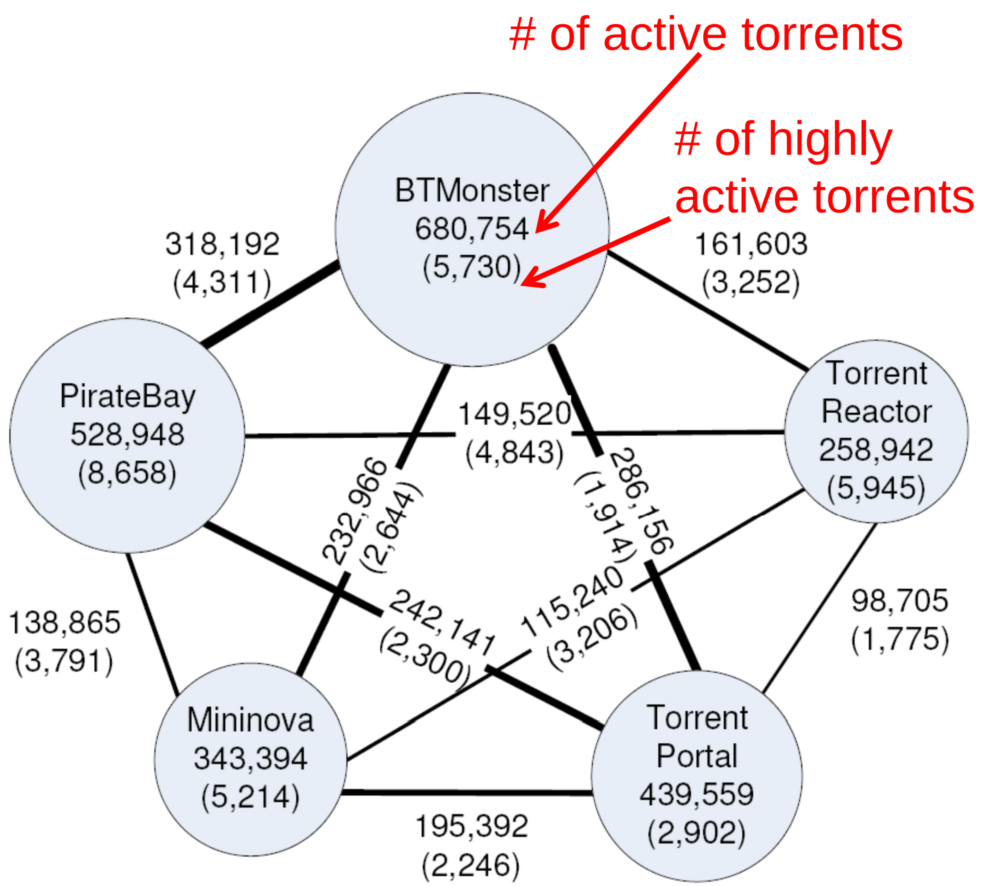
- BitTorrent ecosystem
  - The most successful open app of the decade
- Skype
  - The most successful VoIP app
- ppStream
  - The most successful IPTV app

BitTorrent Ecosystem

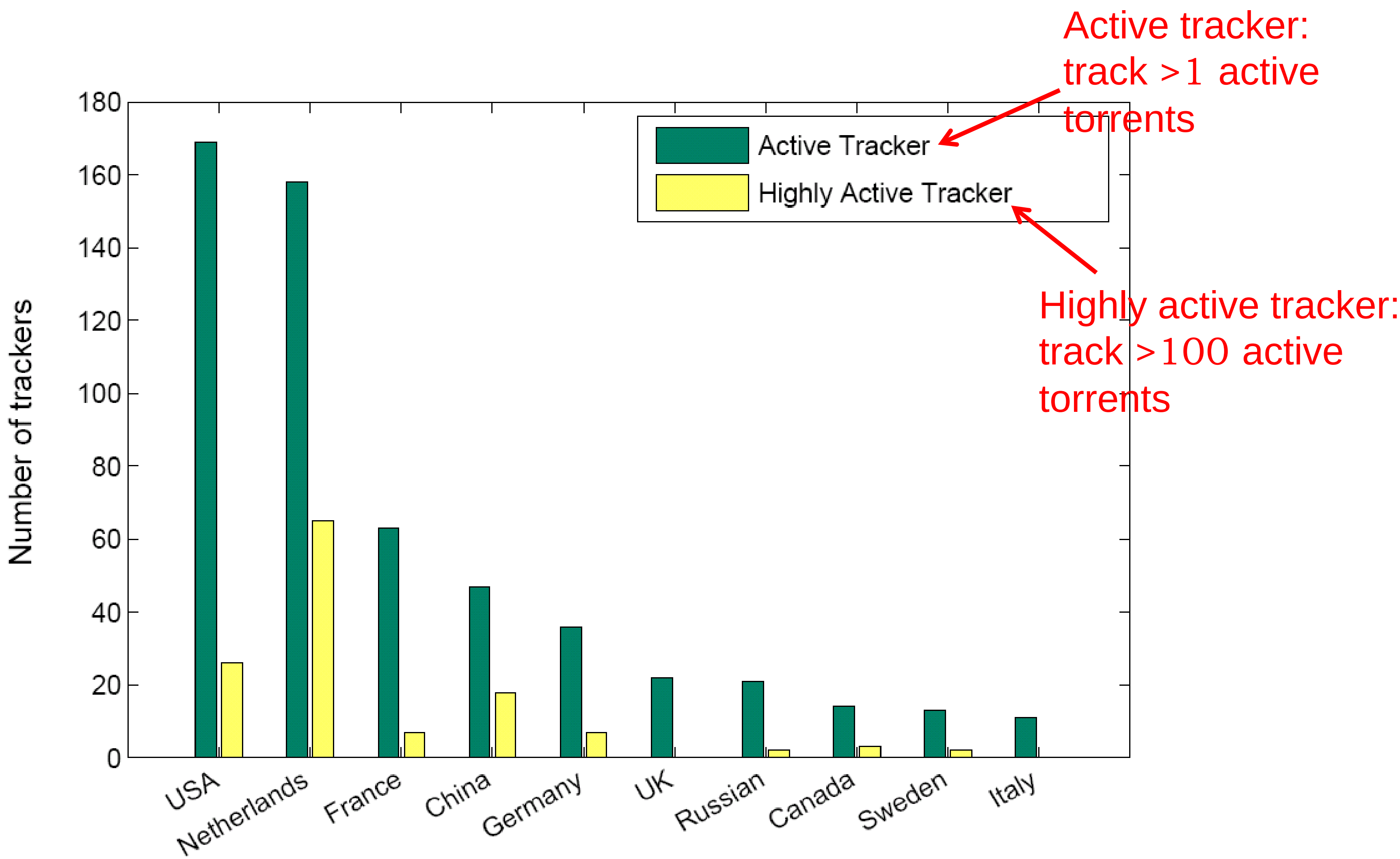


NYU·poly
POLYTECHNIC INSTITUTE OF NYU

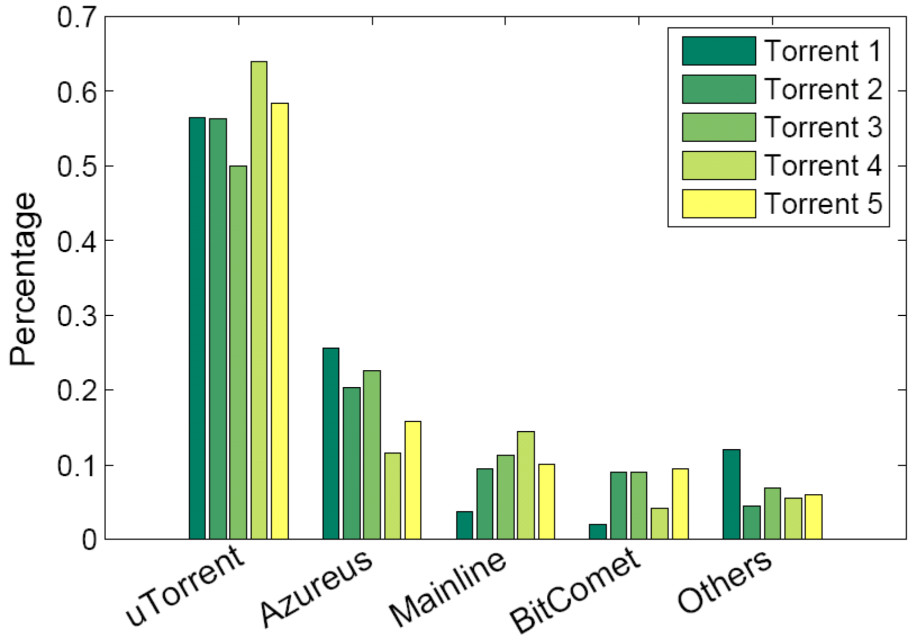
Overlap in Torrent Indexing

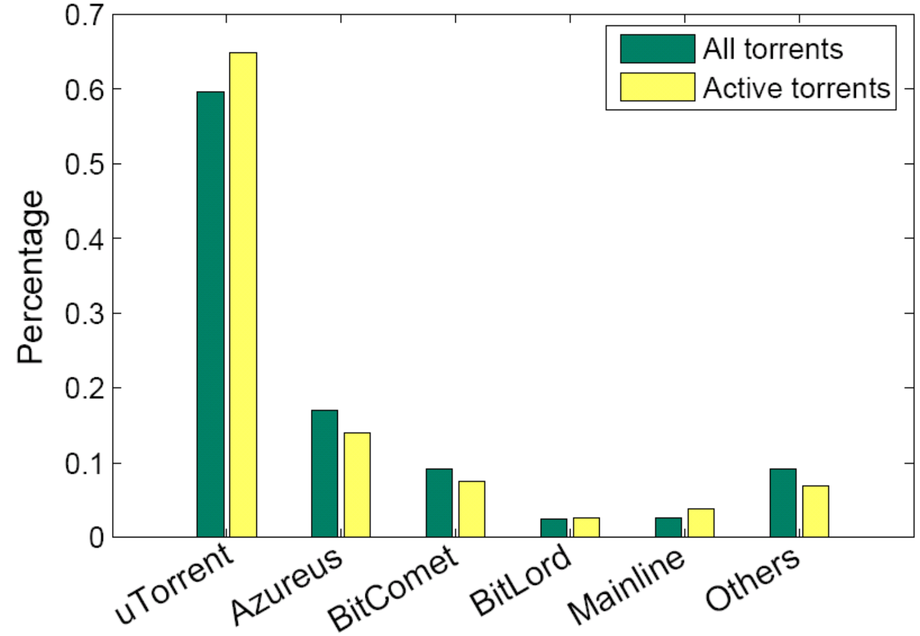


Tracker Distribution



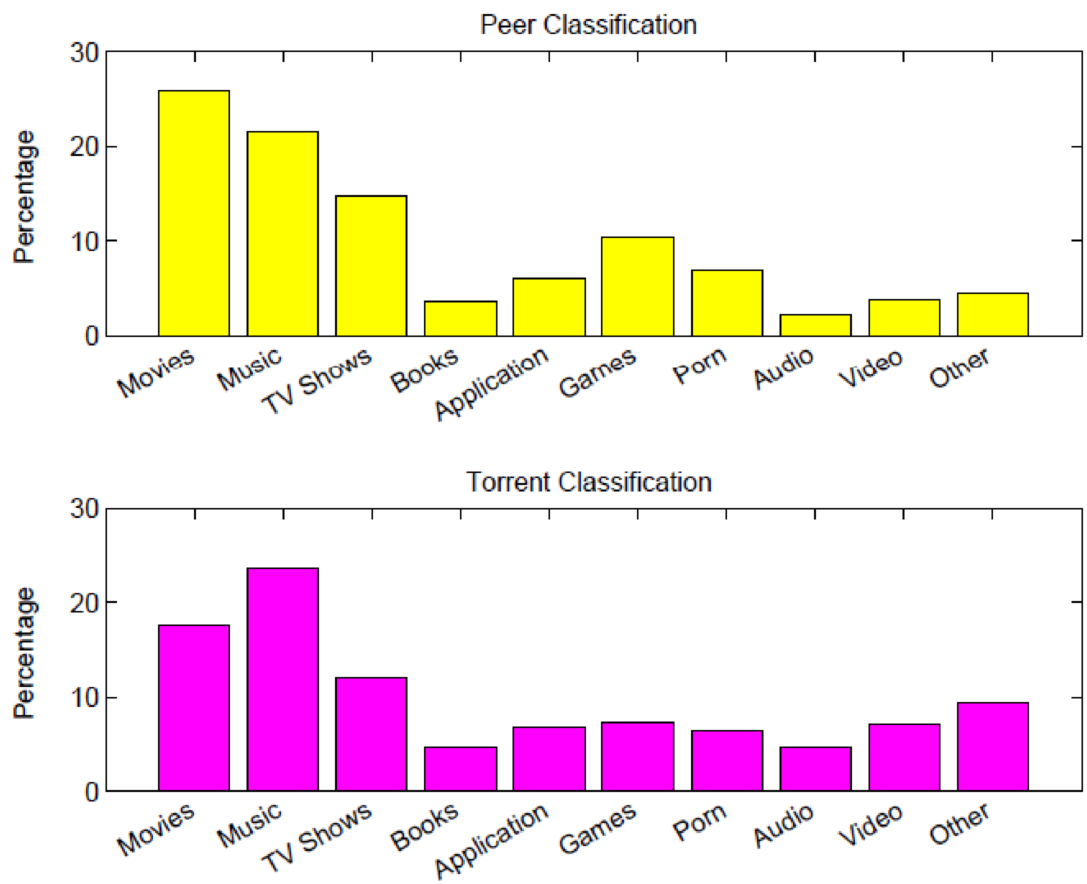
Client Type Popularity





uTorrent & Azurues also form independent DHTs

Content Classification



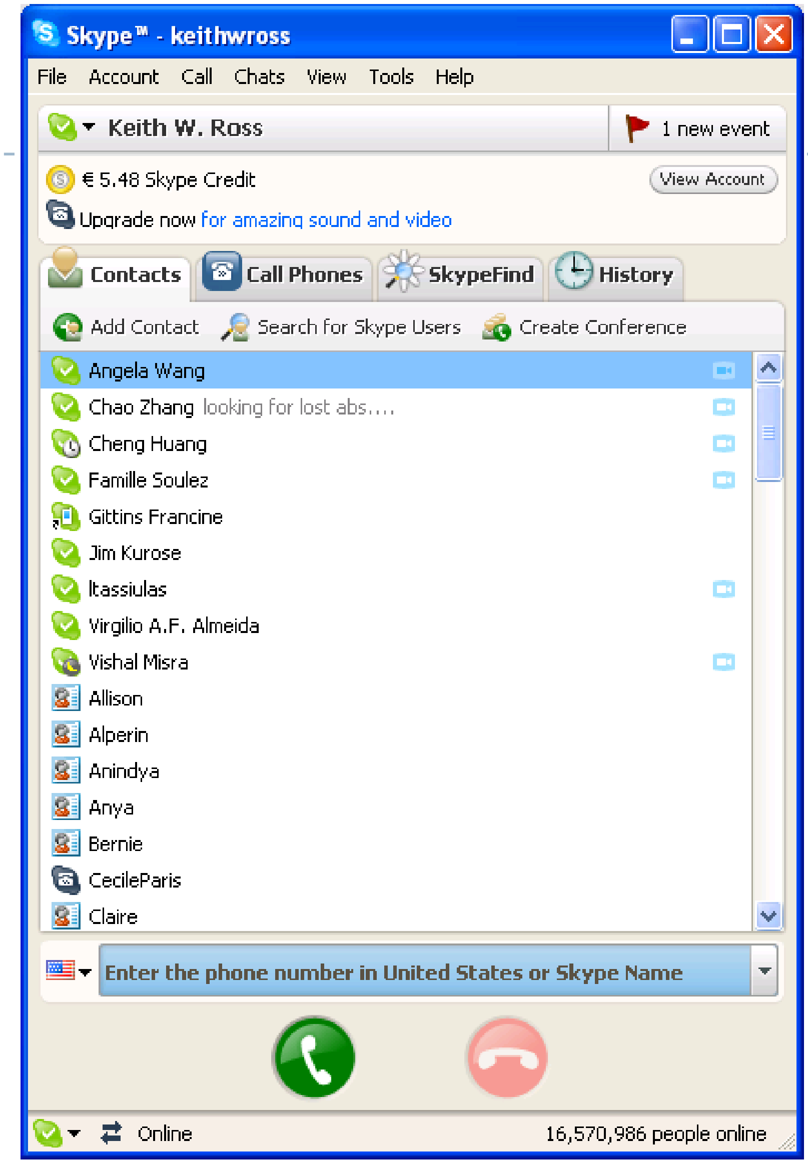
Skype

Minimal infrastructure

- P2P user location
- P2P NAT traversal
- 16M concurrent users

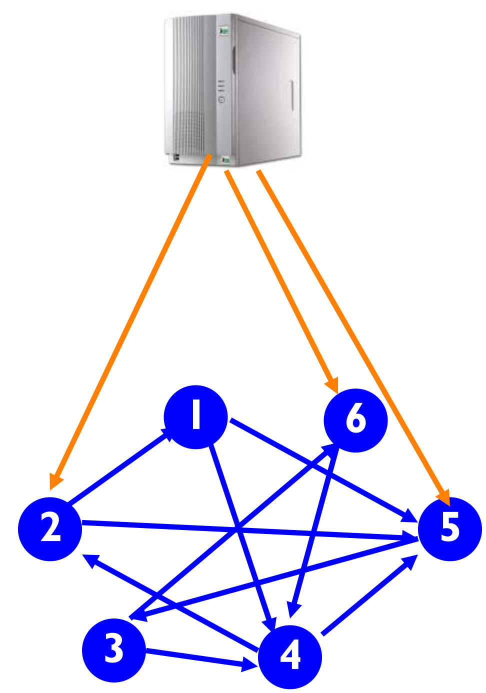
Services

- PC-PC phone
- PC-phone
- Video
- Conferencing



Peer-Assited Video Streaming

- Peers redistribute video <u>chunks</u> to each other (similar to BitTorrent)
  - utilize peer upload capacity
  - reduces load on server
- Large scale deployments on Internet
  - thousands of live/on-demand channels
  - millions of world-wide users daily
- Leading P2P Video Companies
  - CoolStreaming
  - PPStream
  - PPLive
  - Sopcast
  - UUSee



NYU·poly
POLYTECHNIC INSTITUTE OF NYU

CoolStreaming

- The First P2P Video System that attracts 1+ million users
- Shutdown in Jun 10, 2005 due to copyright issues.
- Base technology for Roxbeam Corp., which launched live IPTV programs jointly with Yahoo Japan in October 2006.
- [Infocom05] Xinyan Zhang, Jiangchuan Liu, Bo Li, Tak-Shing Peter Yum, CoolStreaming/DONet: A Data-driven Overlay Network for Efficient Live Media Streaming, In Proceedings of IEEE INFOCOM 2005

PPLive (http://www.pplive.com)

One of the Largest P2P Video Systems in the World

Developed by Xin Yao (HUST, China) in 2004.

85+ Million Users by 2008

Around 800 Channels

Total Num of Channels (788)

PPStream (http://www.pps.tv)

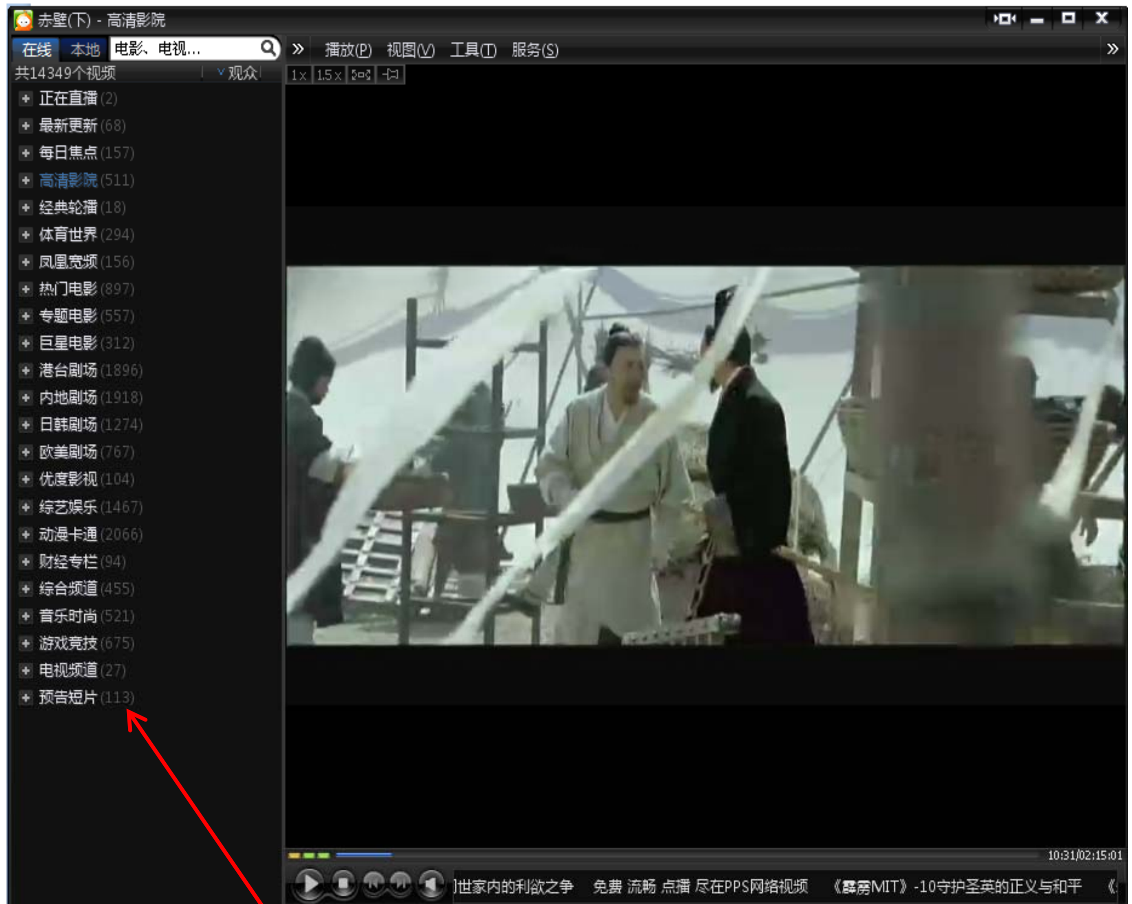
#1 P2P Video System in the World

Developed by Liang Lei and Hongyu Zhang (China) in 2005.

350M installations

~12 Million active users each day

Thousands of channels



Num of Channels

NYU-poly POLYTECHNIC INSTITUTE OF NYU

Some P2P success stories

- BitTorrent ecosystem
  - The most successful <u>open</u> app of the decade
- Skype
  - The most successful VoIP app
- ppStream
  - The most successful IPTV app

Second-Generation P2P Live Streaming

Keith Ross

Polytechnic Institute of NYU

Today's Talk

- Overview of P2P Video Streaming
- View-Upload Decoupling (VUD): A Redesign of P2P Video Streaming
- Queuing Models for P2P Streaming
- LayerP2P: P2P Live Streaming with Layered Video

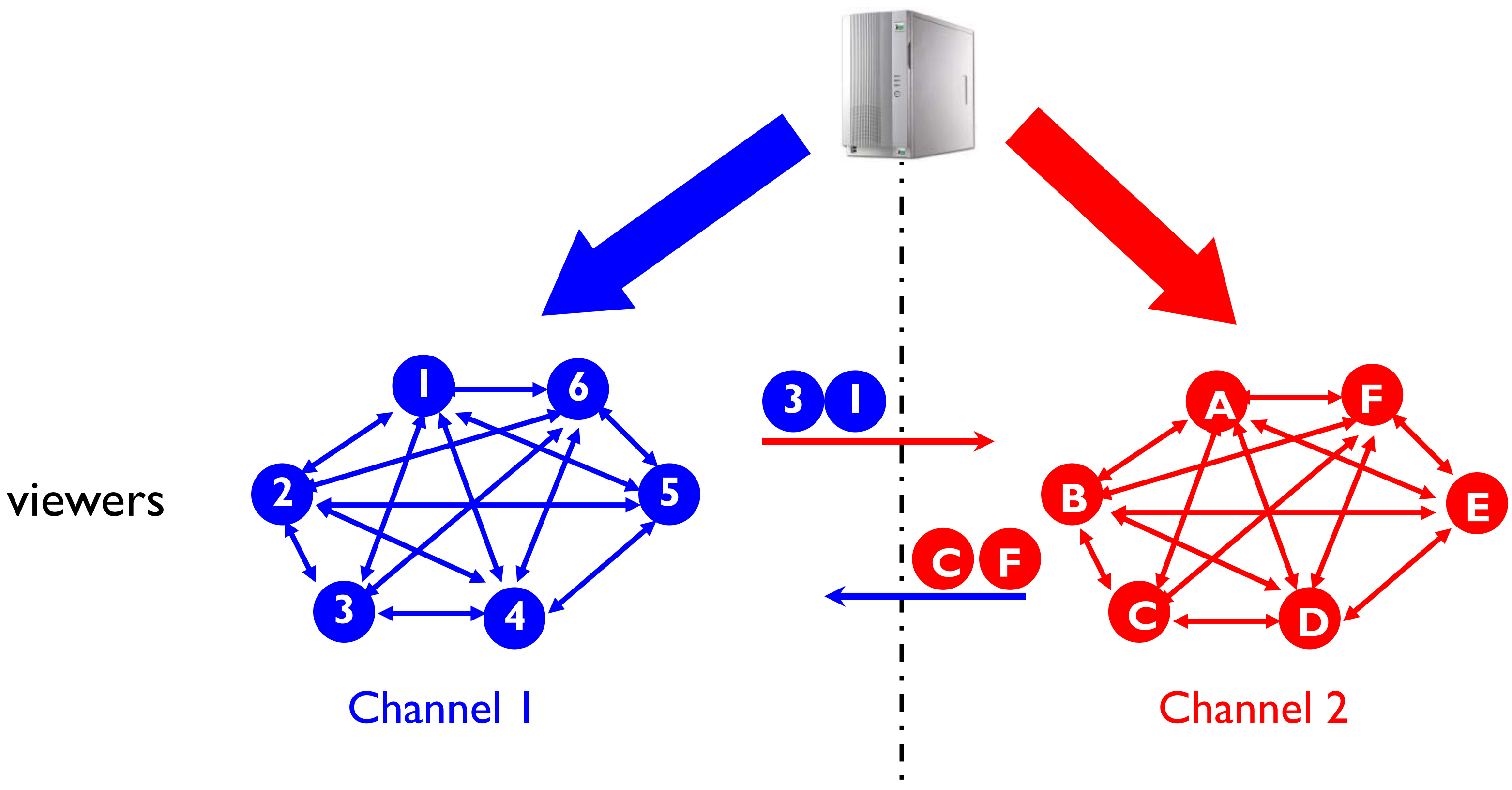
Common features of P2P video streaming

- Multiple Channels
  - Channel Churn
- Heterogeneous Streaming Rates
  - HDTV Channels, VCR-quality channels,…
- Heterogeneous Channel Popularities
  - Very few viewers in less popular channels.
- Isolated Channel Design: ISO
  - Viewer only redistributes channel it is viewing

Problems of Traditional ISO Design

- Large Channel Switching Delay
  - Existing P2P video systems: 10-60 seconds
- Large Playback Lag
  - Existing P2P video systems: 5-60 seconds
- Poor Small-channel Performance
  - Inconsistent and poor performance in small channels.
- *Root causes: channel churn and resource imbalance*

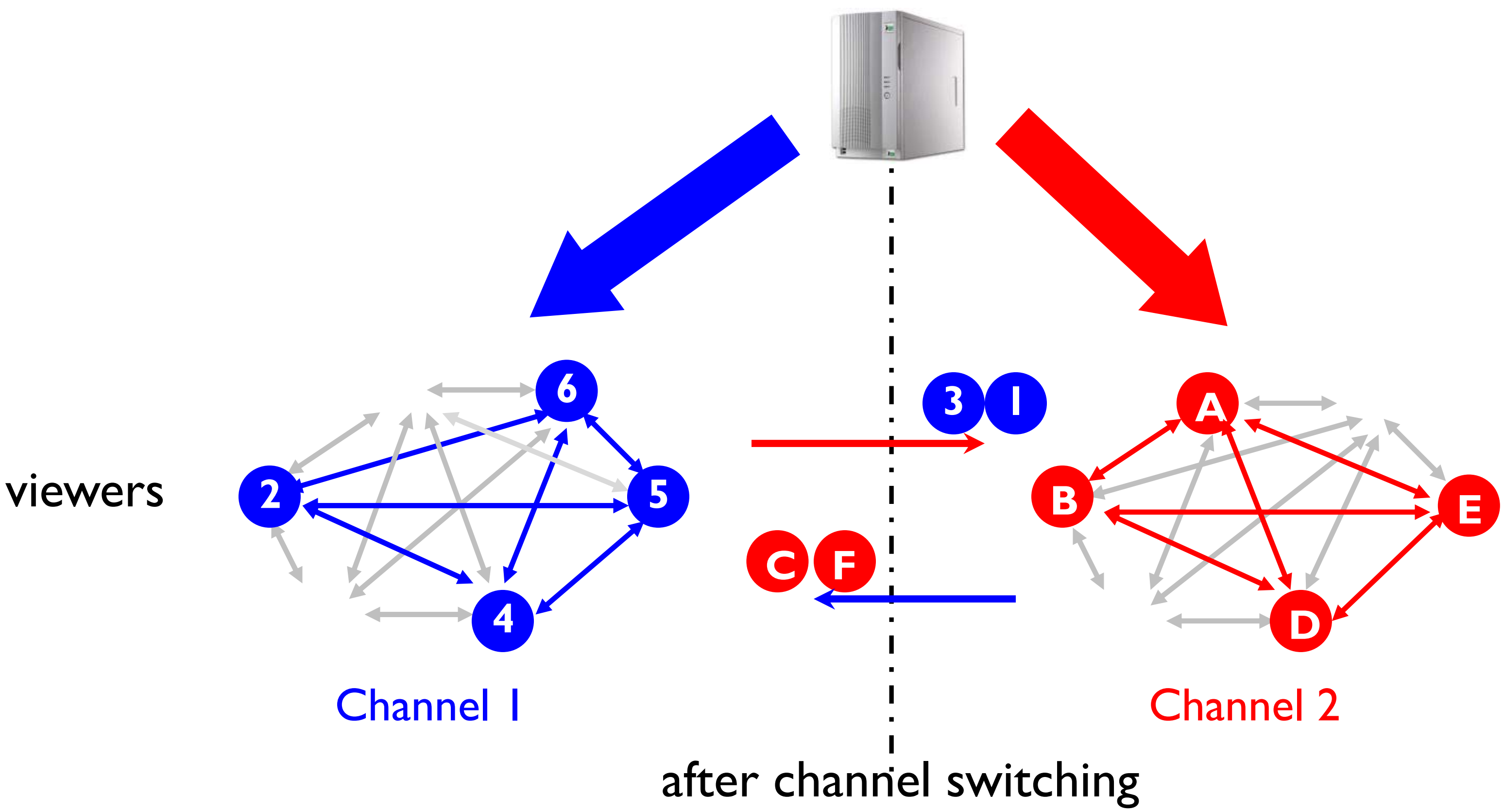
Channel Churn in ISO Design



NYU·poly POLYTECHNIC INSTITUTE OF NYU

Channel Churn in ISO Design

Drawback: distribution systems disrupted when peers switch channels



Resource Imbalance in ISO Design

- Instantaneous resource index for a channel of rate r with n viewers:

$$\sigma = \frac{u_s + \sum_{i=1}^{n} u_i}{nr}$$

- Ratio of available upload rate to required download rate
  - Channel in trouble if $\sigma < 1$

- Resource index can be imbalanced across channels

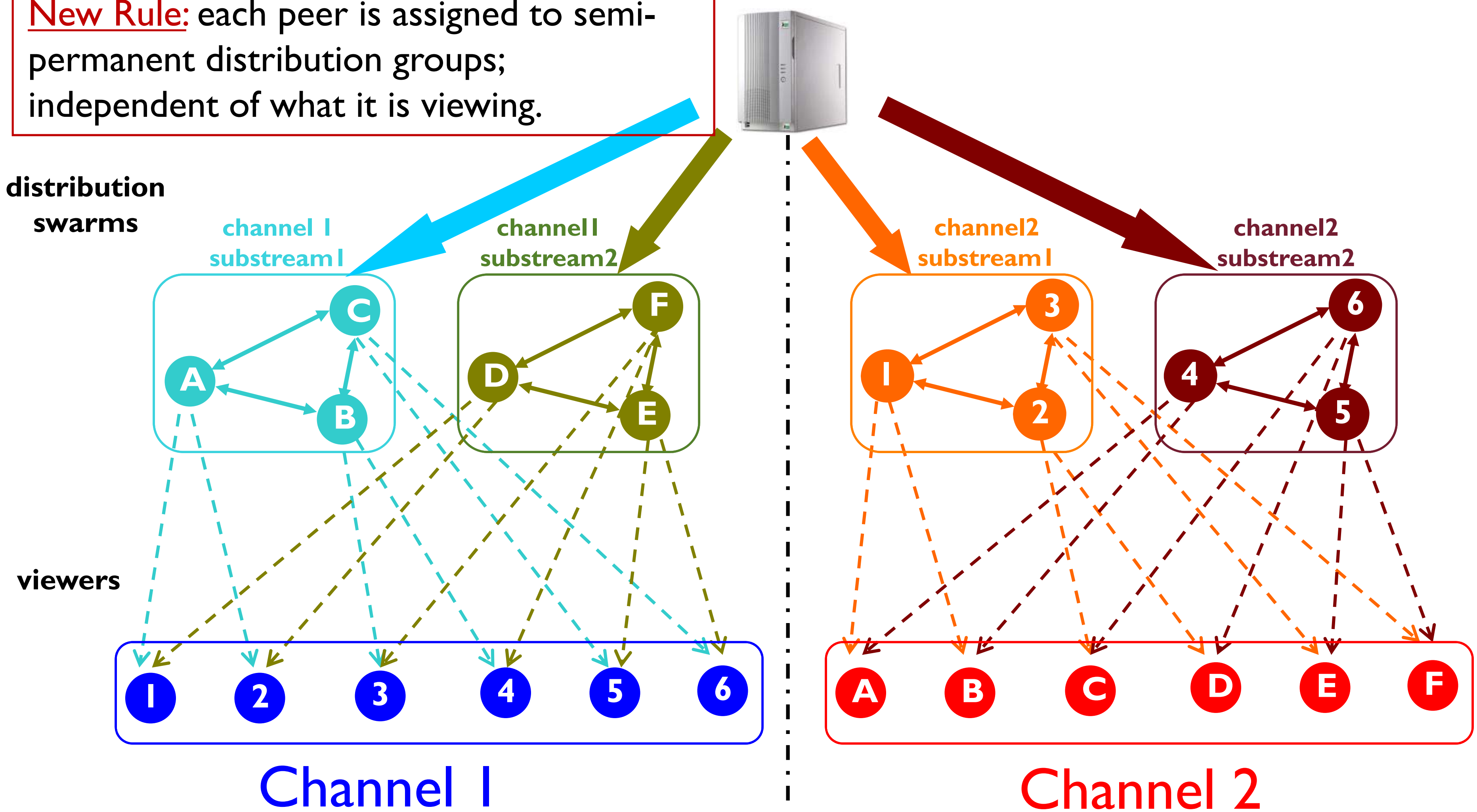
Today's Talk

- Overview of P2P Video Streaming
- View-Upload Decoupling (VUD): A Redesign of P2P Video Streaming
- Queuing Models for P2P Streaming
- LayerP2P: P2P Live Streaming with Layered Video

NYU·poly
POLYTECHNIC INSTITUTE OF NYU

A Redesign of Multi-Channel System: View-Upload Decoupling (VUD)

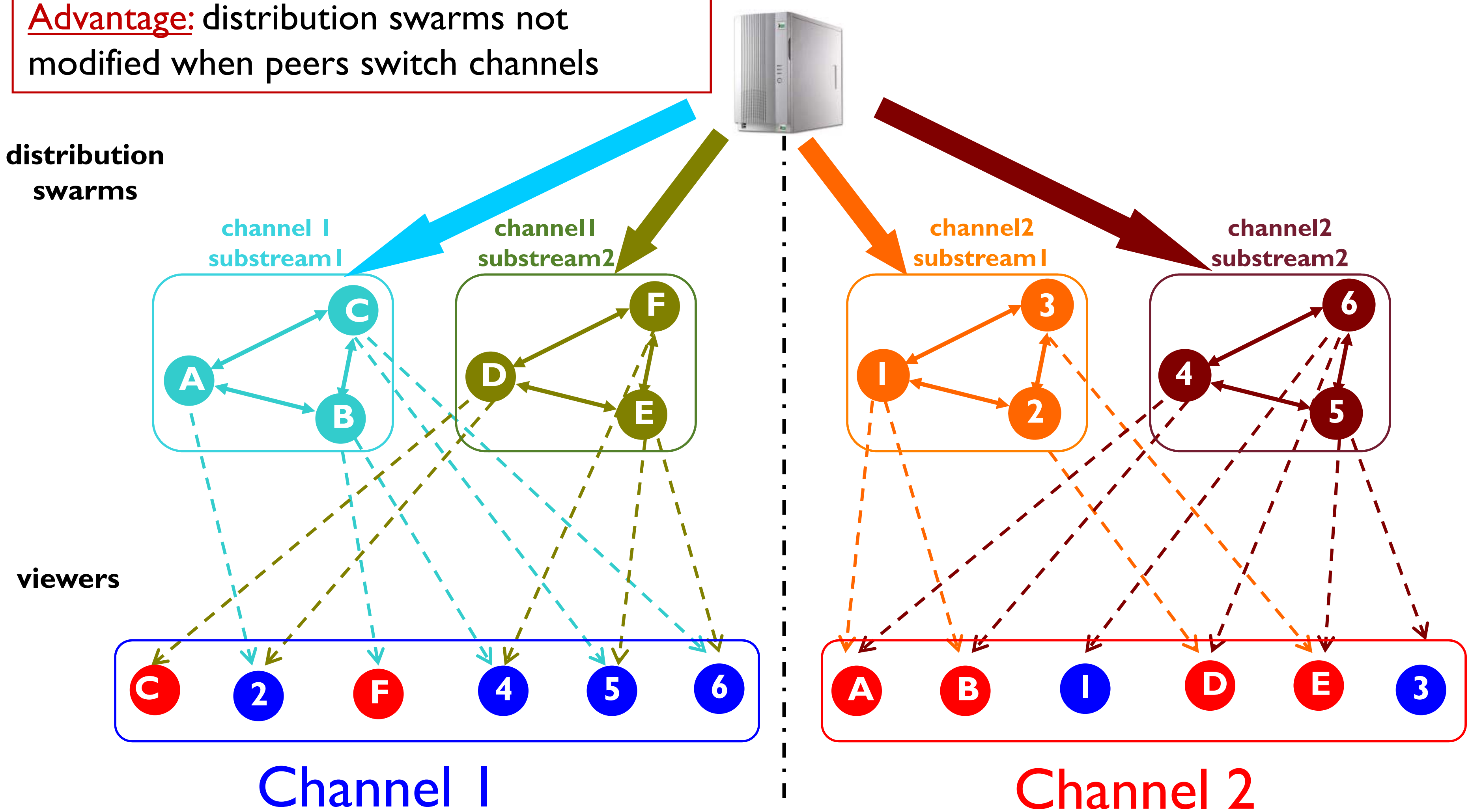
New Rule: each peer is assigned to semi-permanent distribution groups; independent of what it is viewing.



NYU·poly POLYTECHNIC INSTITUTE OF NYU

A Redesign of Multi-Channel System: View-Upload Decoupling (VUD)

Advantage: distribution swarms not modified when peers switch channels



after channel switching

Advantages of VUD design

- Channel Churn Immunity
  - Distribution swarms unaffected by channel churn
- Cross-Channel Provisioning
  - Distribution swarms can be provisioned and adapted to balance resource indexes across channels
- Structured Streaming
  - Scheduling and routing can be optimized within the stable VUD swarms

Key Challenges of VUD design

- VUD Overhead
  - In ISO, peer only downloads video it is watching.
  - In VUD, each peer downloads its assigned substreams as well as the video it is watching.
  - Solution: substreaming

- Adaptive Peer Assignment
  - Bandwidth allocation
  - Peer reassignment

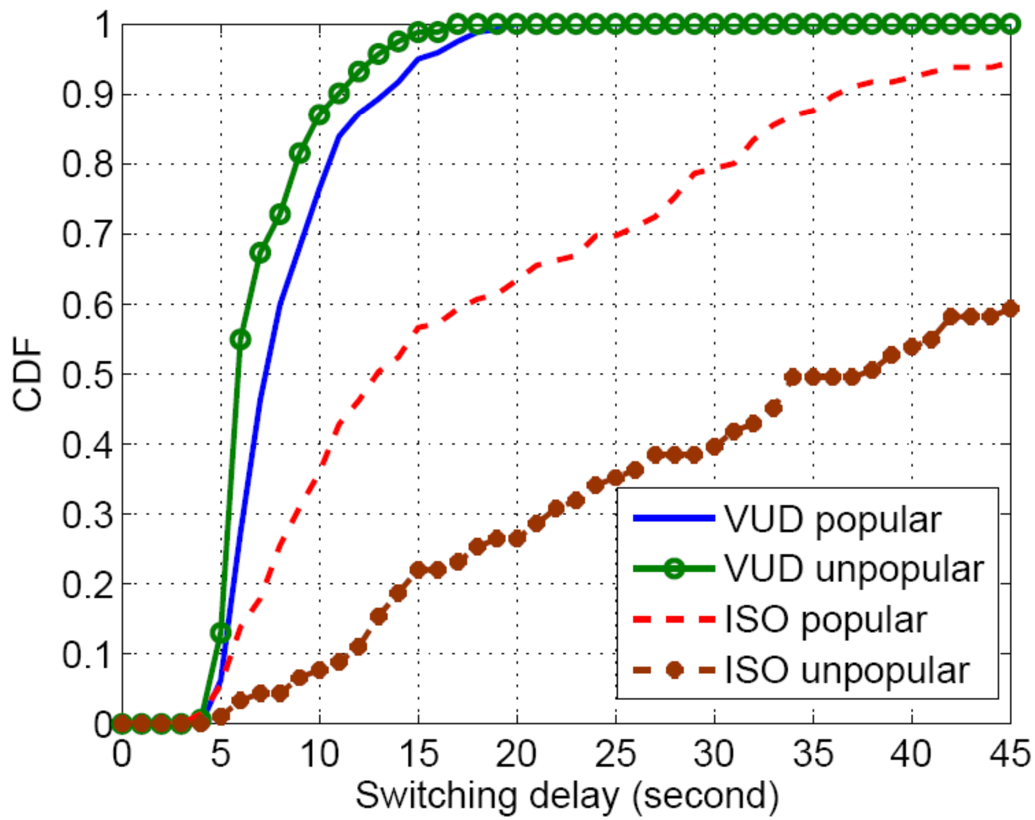
Simulation Experiments

- Simulated features:
  - Channel switching
  - Peer churn
  - Heterogeneous upload bandwidth
  - Packet-level transmission
  - End-to-end latency
  - Zipf-like channel popularity
- Comparison
  - ISO: using Push-Pull scheduling
  - VUD: using Push-Pull scheduling

Simulation Parameters

- 50 channels
  - Video rate 400 kbps each channel
  - Server rate 1 Mbps for each channel
- 2,000 peers
  - Peer upload rates 128-768 kbps
  - Avg peer system time: 67 minutes
  - Channel churn follows IPTV study
- 5 substreams per channel

Channel Switching Delay

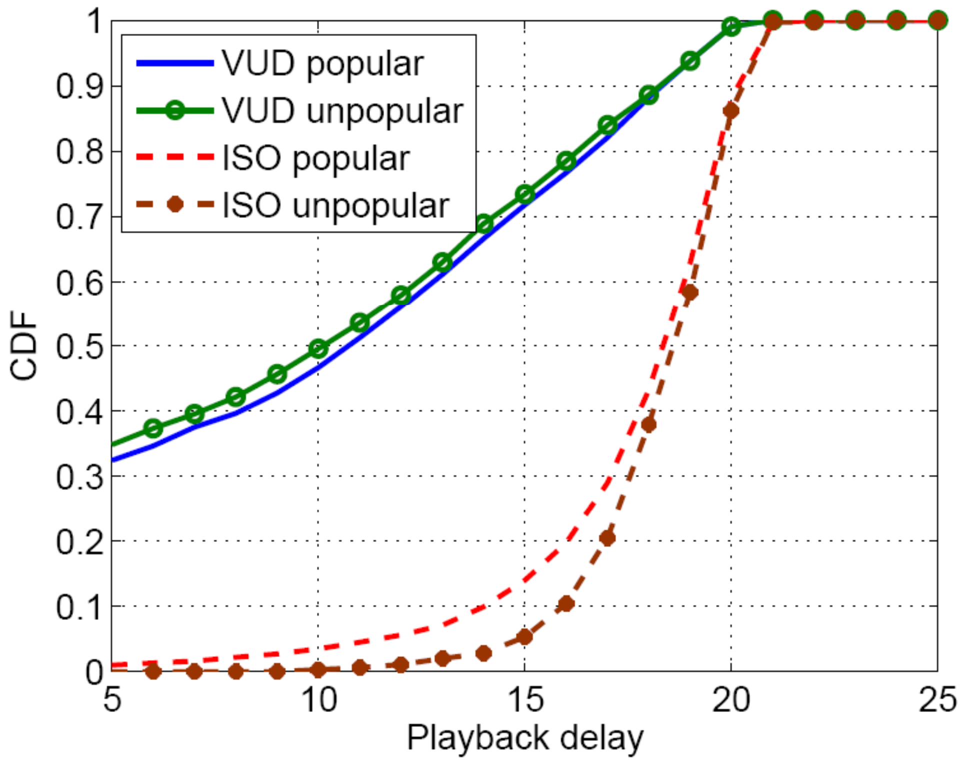


Switching delay = time to acquire 5 seconds of new channel

➢ VUD achieves smaller channel switching delay.

NYU·poly POLYTECHNIC INSTITUTE OF NYU

Playback Lag



➢ VUD achieves smaller playback lag.

Today's Talk

- Overview of P2P Video Streaming
- View-Upload Decoupling (VUD): A Redesign of P2P Video Streaming
- Queuing Models for P2P Streaming
- LayerP2P: P2P Live Streaming with Layered Video

Motivation

- Develop an analytical framework for multi-channel P2P live video systems.

- Use model to study how to optimize VUD performance

- PS = probability of universal streaming
  = fraction of time resource index > 1 for ALL channels

Queuing Network Model

- Each channel can be thought of as a queue
- Each viewer as a customer
- When viewer changes channels, routed to new queue
- Customers move about channels independently:
  - infinite server queues
- Let p_{ij} is probability of switching channel i to j. $P = [p_{ij}]$
- Let $1/\mu_j$ average sojourn time in channel j

- *Can do all kinds of cool things with this model!*
  - *Inspiration from the queuing and loss network literature.*

Closed Queuing Network Model

- Peers never leave (e.g., set-top box peers)
- Now just apply the standard closed Jackson network theory
- Traffic equation $\lambda = \lambda P$
- Relative channel popularity: $\rho_j = \lambda_j / \mu_j$

$$P(M_1 = m_1, \ldots, M_J = m_J) = n! \frac{\rho_1^{m_1}}{m_1!} \cdots \frac{\rho_J^{m_J}}{m_J!}$$

- n is the total number of peers
- M_j = # of viewers in channel j.

Open Queuing Network Model

- Applicable for systems with Peer Churn
- Peers arrive at constant rate and join channel j with prob p_{0j}
- Peer leaves system with probability p_{j0}.

$$P(M_1 = m_1, \ldots, M_J = m_J) = \prod_{j=1}^{J} \frac{\rho_j^{m_j} e^{-\rho_j}}{m_j!}$$

- In this talk, we focus on *Closed Queuing Network Model*.

Analysis of VUD Design

- *Resource Index for substream s of channel j*

$$\sigma_j^s(M_j) = \frac{v_j^s + \sum_{i \in \mathcal{N}_j^s} u_i - n_j^s r_j^s}{M_j r_j^s}$$

M_j = # of viewers in channel j

- Probability of system-wide universal streaming

$$PS = P(\sigma_j^s(M_j) \geq 1,\ s = 1, \ldots, S_j,\ j = 1, \ldots, J)$$

$$PS = P(M_j \leq \delta_j,\ j = 1, \ldots, J\} = \sum_{\mathbf{m} \in \mathcal{M}} n! \frac{\rho_1^{m_1}}{m_1!} \cdots \frac{\rho_J^{m_J}}{m_J!}$$

where $\delta_j = \min_{1 \leq s \leq S_j} \lfloor \frac{v_j^s + \sum_{i \in \mathcal{N}_j^s} u_i - n_j^s r_j^s}{r_j^s} \rfloor$

and $\mathcal{M} = \{(m_1, \ldots, m_J) : m_1 + \cdots + m_J = n,\ 0 \leq m_j \leq \delta_j\}$

NYU·poly
POLYTECHNIC INSTITUTE OF NYU

Asymptotic Analysis of VUD

- How should the VUD groups be dimensioned for large systems?

- Fix number of channels J.
- Let number of peers $n \to \infty$
- Assume for simplicity no substreaming
- Asymptotic regime: $n_j = K_j n$
- How to dimension K_j for large n?

Asymptotic Analysis for VUD

- Initially assume homogenous upload rates: $u_i = u$.
- Critical parameter:

$$\alpha = \sum_{j=1}^{J} \frac{r_j \rho_j}{u - r_j}$$

- Theorem: If $\alpha > 1$, then PS goes to 0 for all choices of K_j. If $\alpha < 1$, then PS goes to 1 if $K_j = r_j \rho_j / \alpha(u - r_j)$

Asymptotic Analysis for VUD

- Heterogeneous peer types: low u^l and high u^h.
- f = fraction of low peers (fixed)
- Can find optimal peer allocations by solving:

$$\text{Maximize} \quad \min_j \{\xi_j K_j^l + \zeta_j K_j^h - \eta_j\}$$

$$\text{Subject to:} \quad \sum_{j=1}^{J} K_j^l = f; \quad \sum_{j=1}^{J} K_j^h = 1 - f$$

- If the value < 0, then PS goes to 0.

Analysis of ISO Design

- Let $\mathcal{M}_j$ be the random set of nodes viewing channel j.

$$PS = P(v_j + \sum_{i \in \mathcal{M}_j} u_i \geq M_j r_j,\ j = 1, \ldots, J)$$

- Once again:

$$P(M_1 = m_1, \ldots, M_J = m_J) = n! \frac{\rho_1^{m_1}}{m_1!} \cdots \frac{\rho_J^{m_J}}{m_J!}$$

- Can be solved used Monte Carlo methods and importance sampling.

Asymptotic Analysis of ISO

- Heterogeneous peer types: low u^l and high u^h.
- f = fraction of low peers (fixed)
- Critical Value:

$$\alpha = \frac{\max_j r_j}{u^l f + u^h (1 - f)}$$

- PS goes to 1 if $\alpha \leq 1$ and goes to 0 otherwise.

Asymptotic Analysis: Example

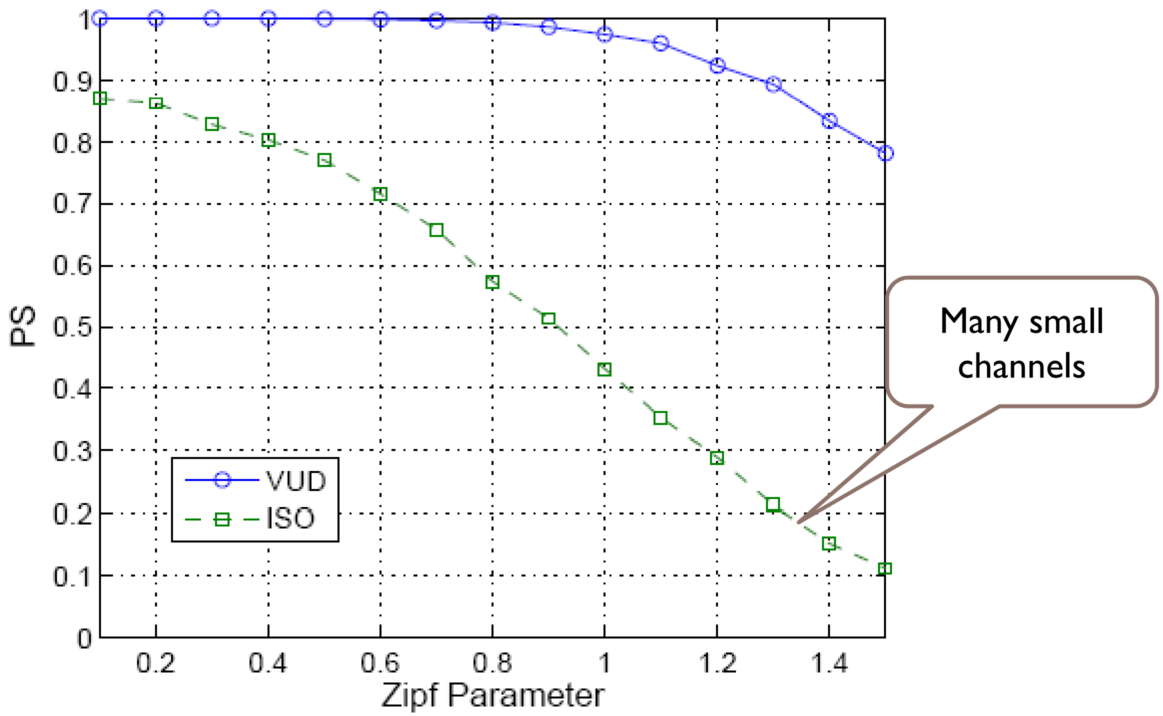
- $u^h = 4r$, $u^l = 2r$, $f = ½$
- $r_1 = 5r$, $r_2 = r$, $\rho_1 = .2$, $\rho_2 = .8$
- ISO: $\alpha > 1$
  - PS goes to 0
- VUD:
  - allocate high-bandwidth peers to channel 1; low bandwidth peers to channel 2.
  - PS goes to 1

Numerical results

- Results from analytical equations
- 1,800 peers
- 20 channels
- $u_l = .2r$ and $u_h = 3r$
- Use asymptotic heuristic to dimension substream swarms

Numerical Results

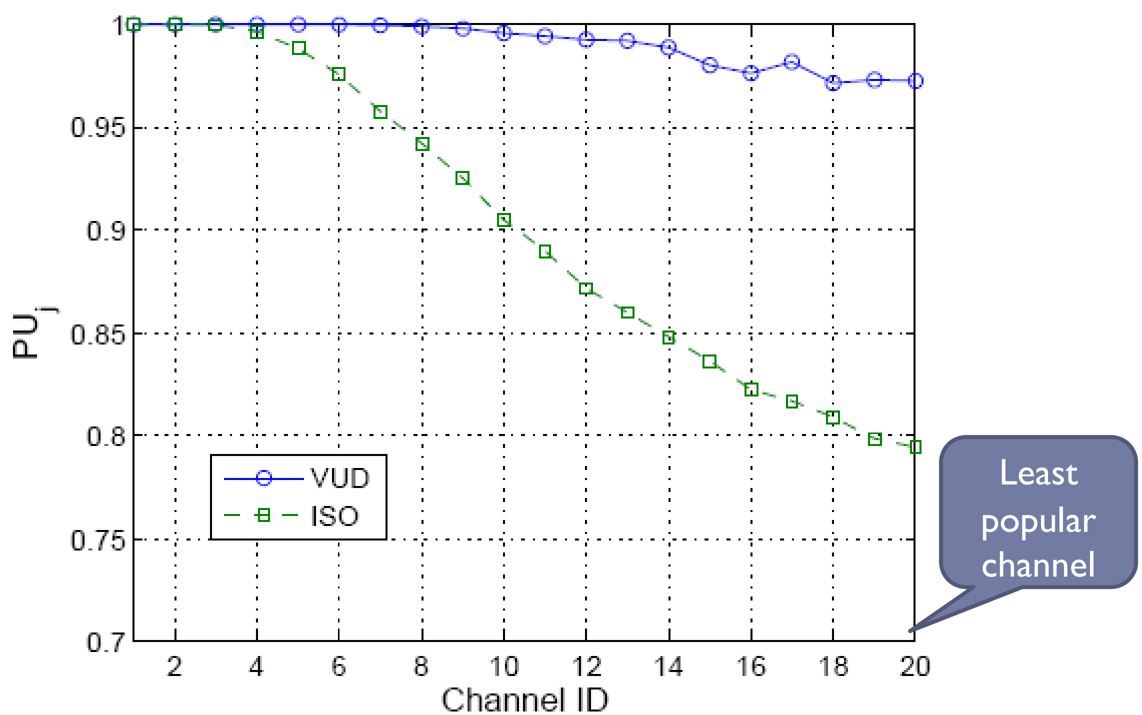
- Probability of System-wide Universal Streaming (PS)
- Vary Zipf parameter



NYU·poly POLYTECHNIC INSTITUTE OF NYU

Numerical Results of VUD Design

- Probability of Universal Streaming in each channel



➢ VUD achieves higher probability of universal streaming (PU_j) in small channels.

NYU·poly
POLYTECHNIC INSTITUTE OF NYU

Refined Heuristic for VUD

- Basic idea: equalize probability of universal streaming across all substreams:

$$P(\sigma_j^s(M_j) \geq 1) = C$$

- Assume normal distribution for M_j
  - Use known mean and variance
- Assume all streams of same rate r

Refined Heuristics for VUD

$$n_j^{s(h)} = \frac{n^h \frac{r}{S}(n\rho_j + C_1\sqrt{n\rho_j(1-\rho_j)} - \frac{v_j^s}{r/S})}{n^h(u^h - \frac{r}{S}) + n^l(u^l - \frac{r}{S})}$$

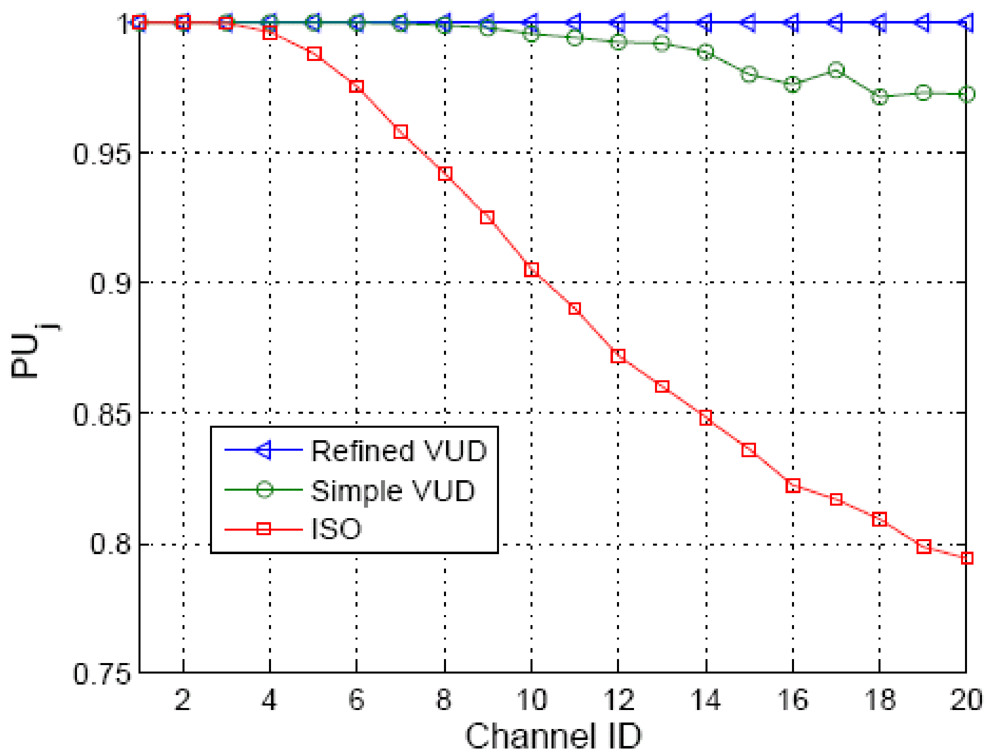
$$n_j^{s(l)} = \frac{n^l \frac{r}{S}(n\rho_j + C_1\sqrt{n\rho_j(1-\rho_j)} - \frac{v_j^s}{r/S})}{n^h(u^h - \frac{r}{S}) + n^l(u^l - \frac{r}{S})}$$

Num. of high-bandwidth peers in substream s of channel j

Num. of low-bandwidth peers in substream s of channel j

Refined Heuristic for VUD Streaming

- Probability of universal steaming in each channel.



➢ Refined VUD can achieve higher probability of universal streaming in small channels.

NYU·poly
POLYTECHNIC INSTITUTE OF NYU

Today's Talk

- Overview of P2P Video Streaming
- View-Upload Decoupling (VUD): A Redesign of P2P Video Streaming
- Queuing Models for P2P Streaming
- LayerP2P: P2P Live Streaming with Layered Video

A BitTorrent Lesson

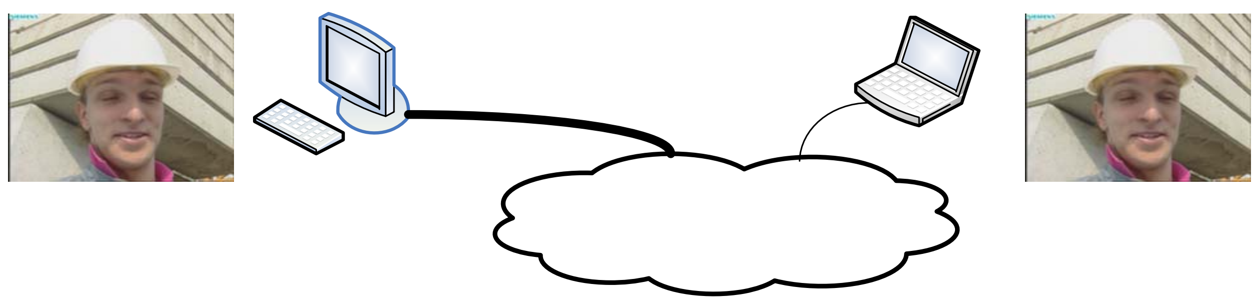
- BitTorrent is successful
  - 50+ client implementations
  - Dozen public trackers
  - 5-10 million users
- Why BitTorrent?

P2P design + **Incentives** → **Resources**

- First generation P2P applications: Gnutella
  - 70% of users are free-riders
- Second generation P2P applications: BitTorrent

Lack of Incentives in P2P Live Streaming

- Some peers contribute much more bandwidth than others
  - In PPLive, an institutional peer may upload 30 times more than a residential peer
- But… they all receive the same video quality
  - Why upload more than tit-for-tat?



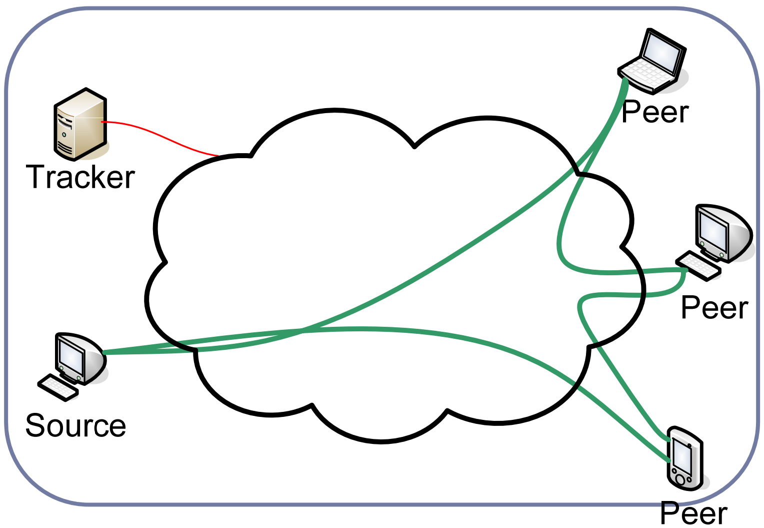
Our Design Philosophy

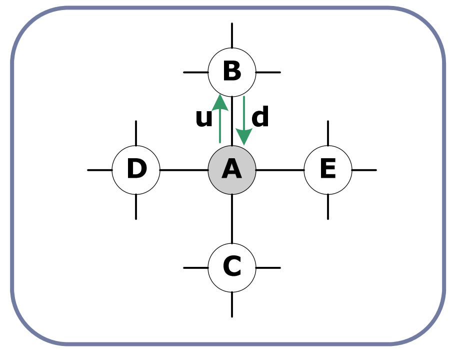
- Bandwidth-rich period
  - Average upload bandwidth > full video rate

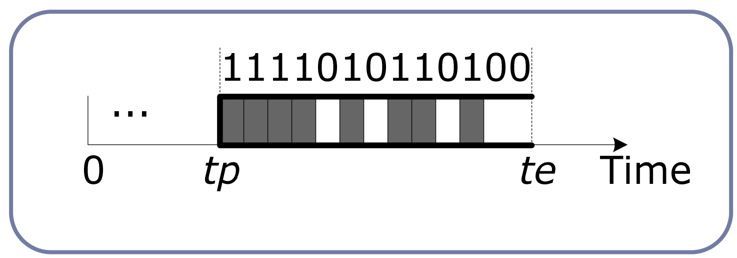
- Bandwidth-deficient period
  - Average upload bandwidth < full video rate
  - More upload contribution → better video quality

System Design I: Chunk-Based Mesh-Pull Design

- Adopted by most existing P2P live streaming systems
- Peers are self-organized into a mesh
- Each chunk will be explicitly identified, requested, and scheduled







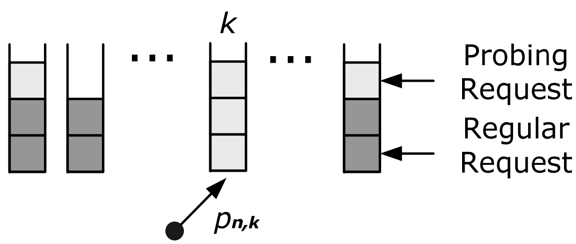
System Design II: Layered Video

- Use layered video to provide differentiated video quality
- Encode a video into multiple layers with nested dependency
  - Base layer provides basic video quality
  - Enhancement layers provide refined video quality

- Properties
  - Comparable video coding efficiency with single-layer video
  - Has been standardized: H.264 SVC
  - Open source real-time codecs: FFmpeg

System Design III: Tit-for-Tat

- Supplier side scheduler
  - A tit-for-tat like strategy
    - If Alice receives a higher download rate from her neighbor Bob, she will allocate a large share of upload bandwidth to Bob
    - Pair-wise proportional bandwidth allocation

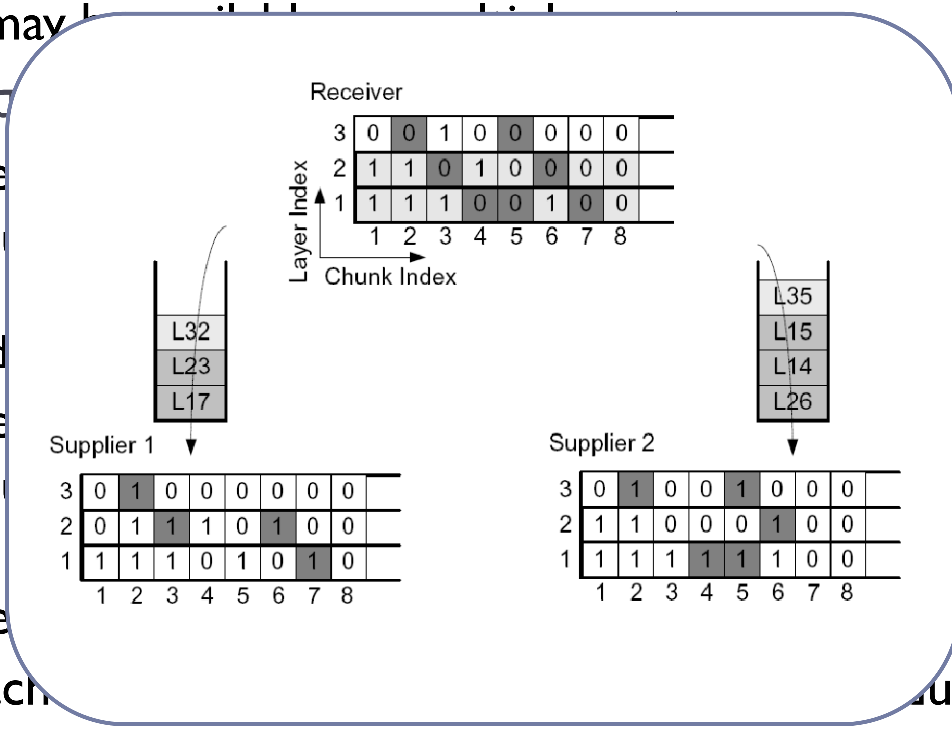


$$p_{n,k} = \frac{I_{n,k}(d_{n,k} + \epsilon)}{\sum_{i \in \mathcal{K}_n} I_{n,i}(d_{n,i} + \epsilon)}$$

  - Upload more → Larger share of upload bandwidth from neighbors → More layers → Better video quality

System Design IV: Prioritized Random Scheduling

- Receiver side scheduler
  - How to request these LCs
    - A receiver may have multiple missing LCs to request
    - Each LC may
  - Regular req
    - Regular re
      - The requ)<Un<R(l+1)
      - Regular pair-wise bandwid
    - Probing re
      - The requ
      - Probing verloaded
    - Regular re ing requests
    - Within each uling



System Design V: Partnership Policy

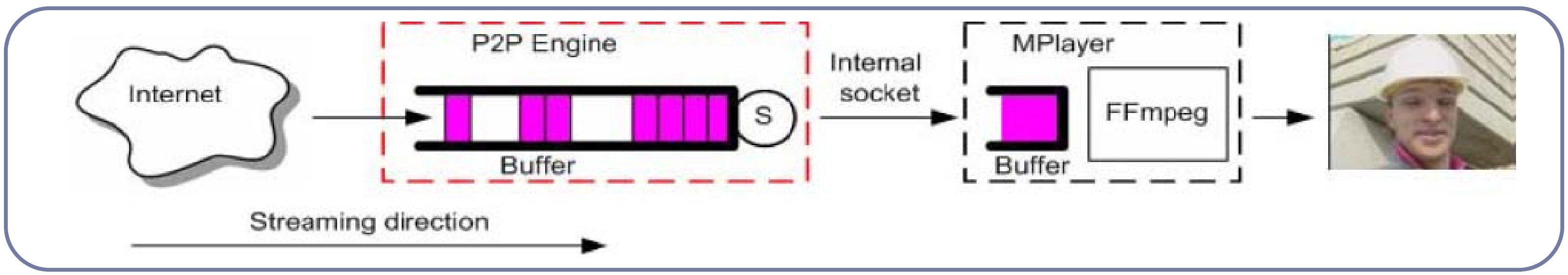
- New partnership
  - Initiator and receptor: If peer A initiates the neighbor establishment with peer B, then peer A is an initiator of peer B, and peer B is a receptor of peer A.
  - Initially, initiator (peer A) allocates a relatively large share of upload bandwidth to receptor (peer B), but receptor (peer B) only allocates a relatively small share of upload bandwidth to initiator (peer A).
  - Similar to BitTorrent's optimistic unchoking
- Partner adaptation
  - Periodically drop the worst partner

Features to Prevent Free-Riding

- Pair-wise bandwidth allocation:
  - Free-rider can only obtain small shares of bandwidth from its partners
- Partner adaptation
  - Free-rider will be dropped by its partners
- Initiator and receptor
  - Free-rider can actively locate a large number of partners, but since it's an initiator, it can only obtain small shares of bandwidth from its partners

System Implementation

- Objectives
  - Demonstrate the viability of the schemes
  - Evaluate the system performance in the Internet
- Approach
  - C++ on Linux
  - Tracker, source, and peer
  - UDP
  - Temporal scalable coding and FFmpeg

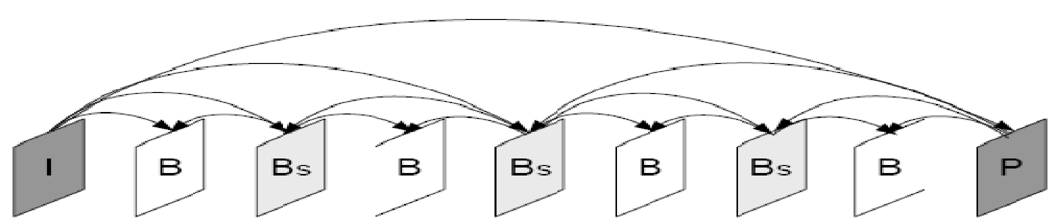


PlanetLab Experiment

- 100+ nodes
  - One tracker, one source, and 100 peers
- Three types of peers under two scenarios

Peers	free-rider	residential	institutional
Upload rate (kbps)	0	400	1000
Underloaded&No Free-Riding	0	40%	60%
Overloaded&Free-Riding	15%	43%	42%

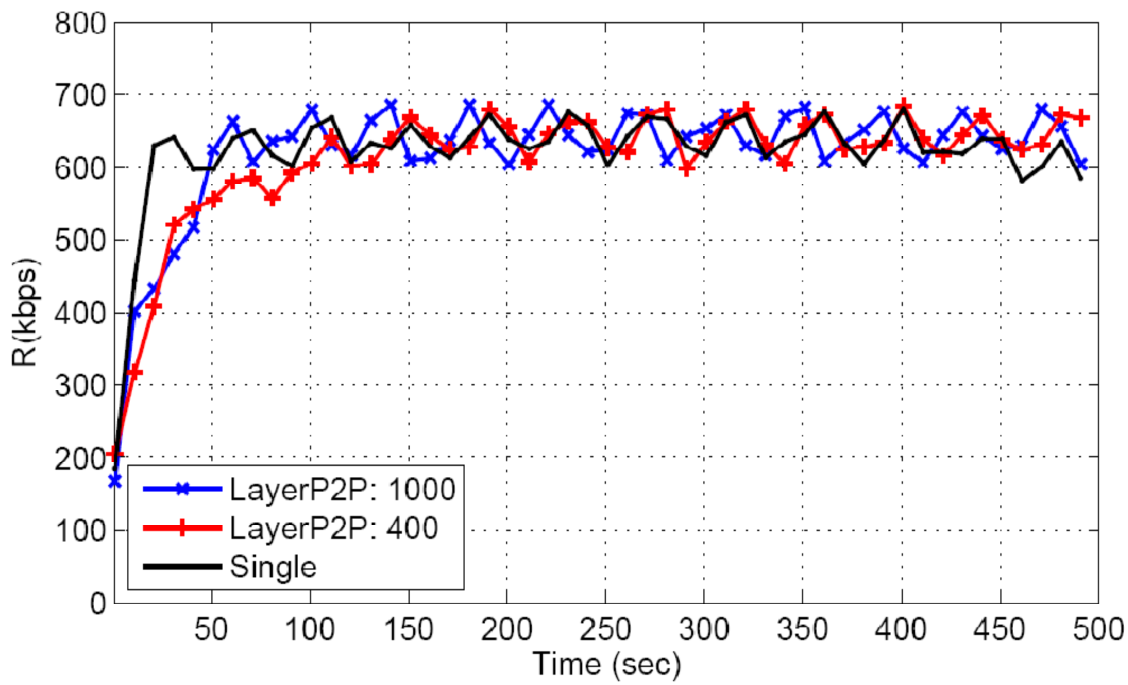
- H.264/SVC temporal scalable video



  - "ICE" sequence, 4CIF (704x576), 30 frames/second
  - 290/230/100 kbps

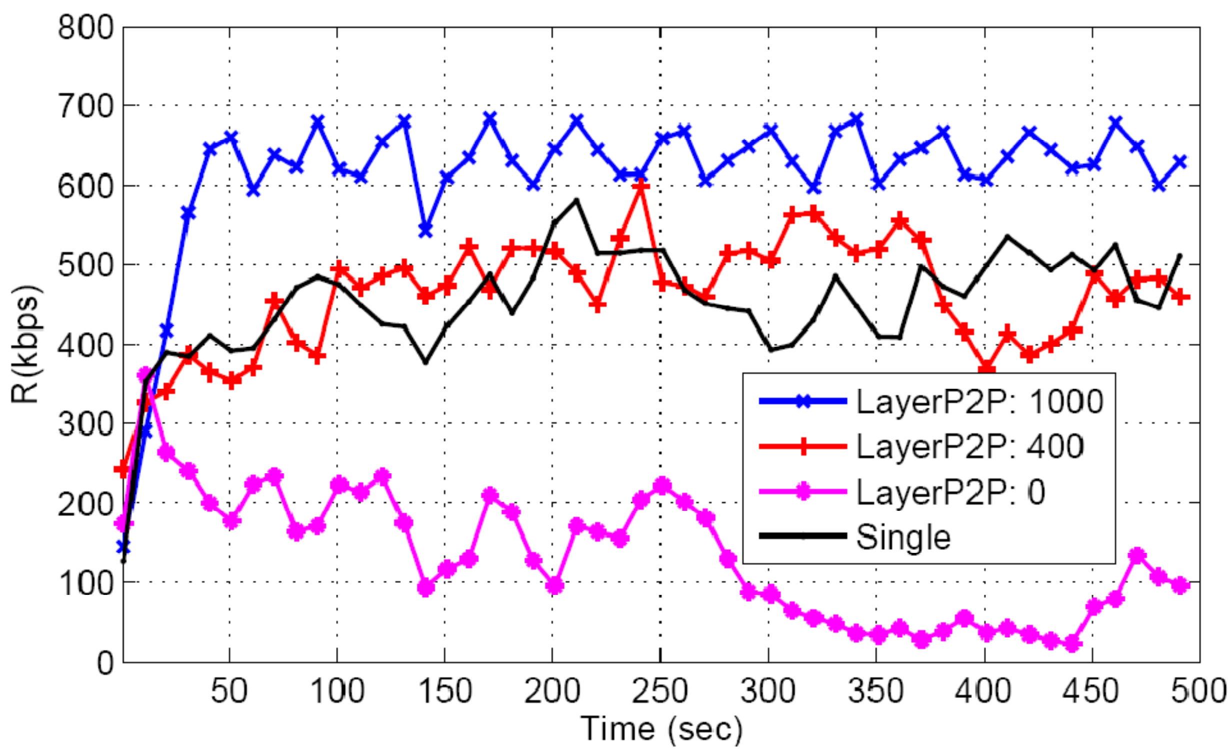
Underloaded System

- Resource index = 1.23
- Trace of received video rate



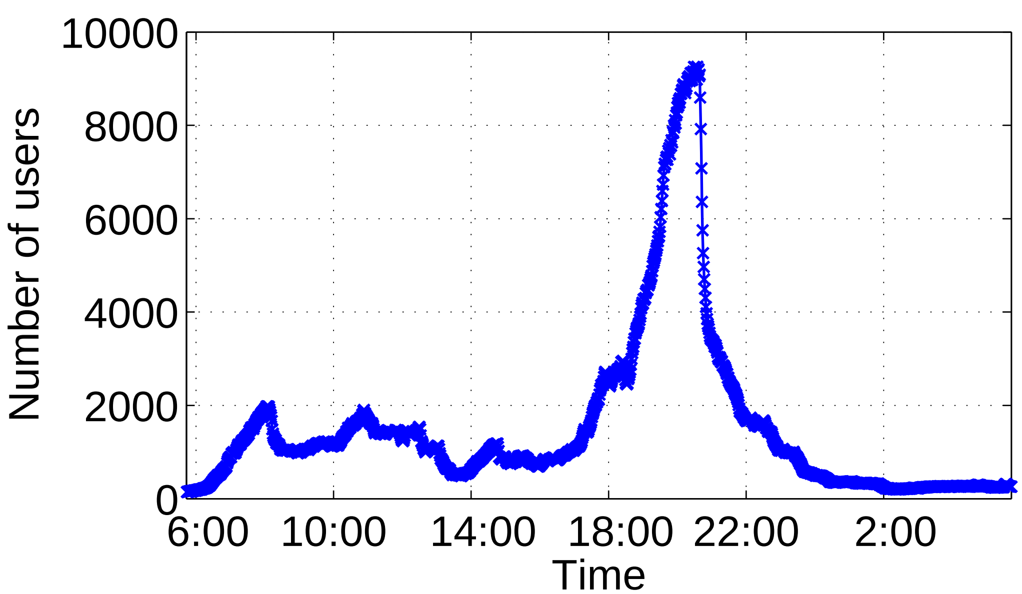
Overloaded System

- Resource index = 0.97
- Trace of received video rate

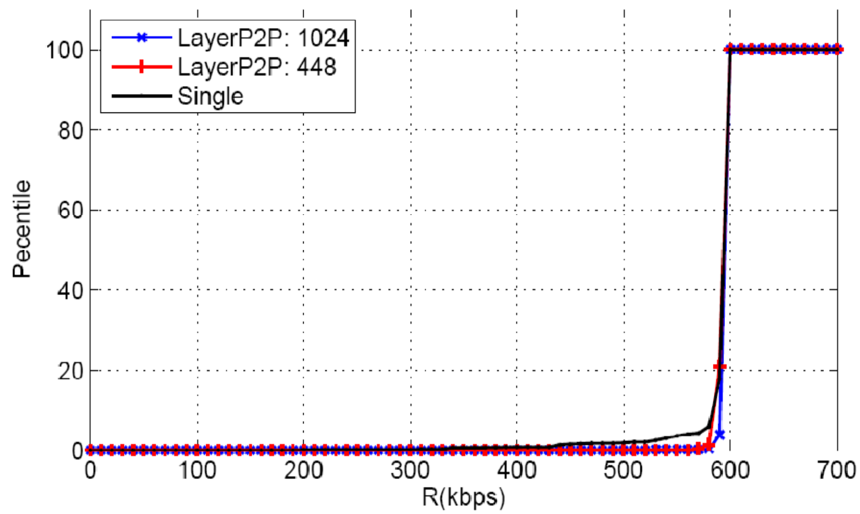


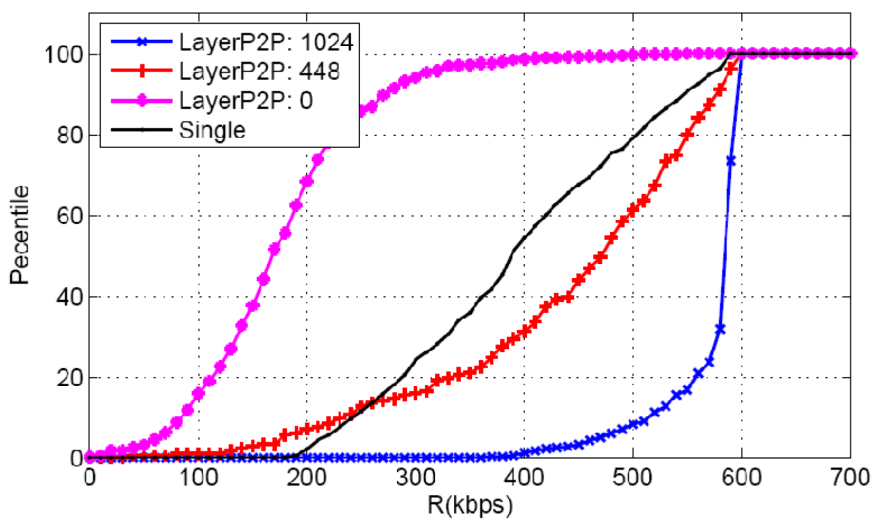
Trace-Driven Simulation

- Objectives
  - Investigate the system performance with real peer dynamics
- Approach
  - 24 hours/100,000 video sessions/Maximum of more than 9,000 simultaneous peers



Simulation Result





- Underloaded system
  - No free-rider
  - Resource index = 1.26
- Overloaded system
  - 30% free-riders
  - Resource index = 0.73

Summary

- Introduced a new design of P2P Video systems: View-Upload Decoupling (VUD)
- Developed a tractable analytic theory to study ISO and VUD streaming
- Introduced a new design of P2P Video systems with built-in incentives: LayerP2P

More Details…

- “View-Upload Decoupling: A Redesign of Multi-Channel P2P Video Systems”,
  Di Wu, Chao Liang, Yong Liu and Keith Ross,
  IEEE Infocom, Mini-conference, 2009.

- “Queuing Network Models for Multi-Channel P2P Live Streaming Systems”,
  Di Wu, Yong Liu and Keith Ross,
  IEEE Infocom, 2009.

- “LayerP2P: P2P Live Streaming with Layered Video”,
  Zhengye Liu, Yanming Shen, Keith Ross, Shivendra S. Panwar and Yao Wang,
  Submitted to IEEE Trans. on Multimedia, 2009 (and related ICNP paper)

- Patent Pending

Thank You !!