Kangaroo: Video Seeking in P2P Systems

Xiaoyuan Yang †, Minas Gjoka ¶, Parminder Chhabra †,
Athina Markopoulou ¶, Pablo Rodriguez †

† Telefonica Research

¶ University of California, Irvine

Outline

- Motivation and contributions
- Related work
- Architecture
  - Peers
  - Tracker
- Experimental evaluation
  - Tracker
  - Segment Scheduler
- Conclusion and future work

Peer-to-peer
Challenges

- File sharing
  - reduces the service provider's cost
  - allows scalability
- Live Streaming
  - reduces upload cost, particularly important for
    - high quality videos, popular events
  - challenge: time sensitivity
- Video on Demand
  - challenge: lack of synchronization between peers
- **Video on Demand with jump operations**
  - **enhances user experience**
  - **additional challenge: constant neighborhood re-adjustments**

Problem statement

- Build a hybrid Video on Demand P2P system that
  - supports jumps
  - provides low buffering times
  - high swarming throughput

  - without overly provisioned peers and without aggressive prefetching

Video on Demand P2P

Related Work

- Prior Work

Bulletmedia - N. Vratonjic, P. Gupta, N. Knezevic, D. Kostic, A. Rowstron, "Enabling DVD-like features in P2P Video-on-Demand Systems", *in ACM P2P-TV Workshop 2007*.

Gridcast - B. Cheng, X. Liu, Z. Zhang, H. Jin, "A Measurement Study of a Peer-to-Peer Video-on-Demand System ", *in IPTPS 2007*.

PPLive – Y. Huang, T.Z. J. Fu, D.M Chiu, K.C.S. Lui, C. Huang, "Challenges, Design and Analysis of a Large-scale P2P VoD System", in Sigcomm 2008

Architecture
Key Design Choices

- Mesh-based P2P system with a pull model.
  - small segment size (64KB)
  - small active set of neighbors
- **Peers**
  - adaptive hybrid segment scheduler
  - neighborhood manager
- **Smart tracker**
  - smart neighbor selection
  - history based neighbor selection

Architecture – Peers (1)

Segment Scheduler

- Segment scheduler decides what segment to download next
  - *greedy* strategy chooses segments for sequential playback
  - *altruistic* strategy chooses local-rarest segments

- We propose a **dynamic** hybrid segment scheduler
  - starts with 80% greedy and 20% altruistic
  - ratio of sequential vs local-rarest segments varies dynamically depending on buffer size and segment deadline

Architecture – Peers (2)

Neighborhood manager + Peer Selection

- Neighborhood manager
  - maintains a "healthy neighborhood'
  - limits the number of active connection.
  - requests segments from "least useful" neighbors

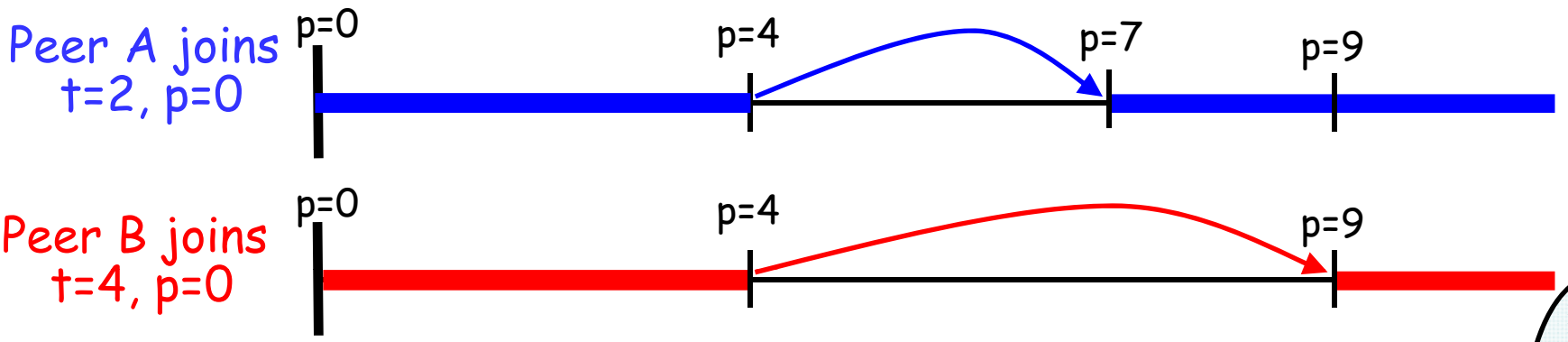
- Dynamic batching of "Have" messages
  - important reduction of control traffic

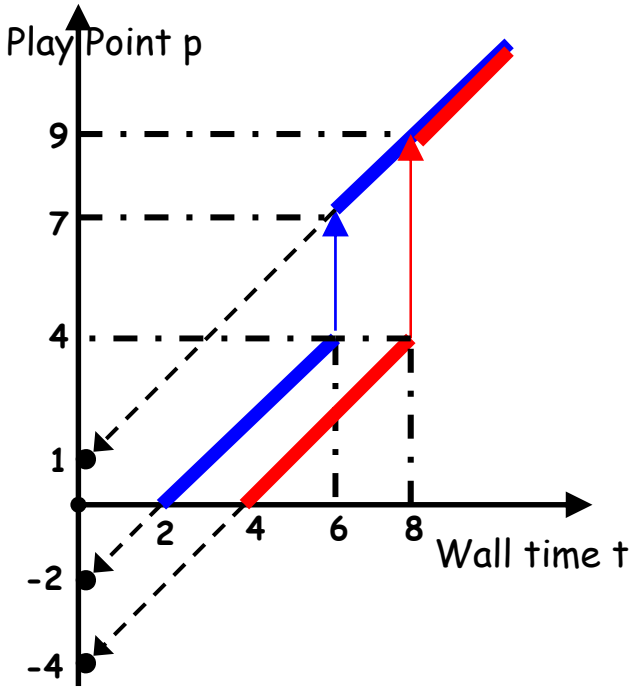
Architecture – Tracker (1)

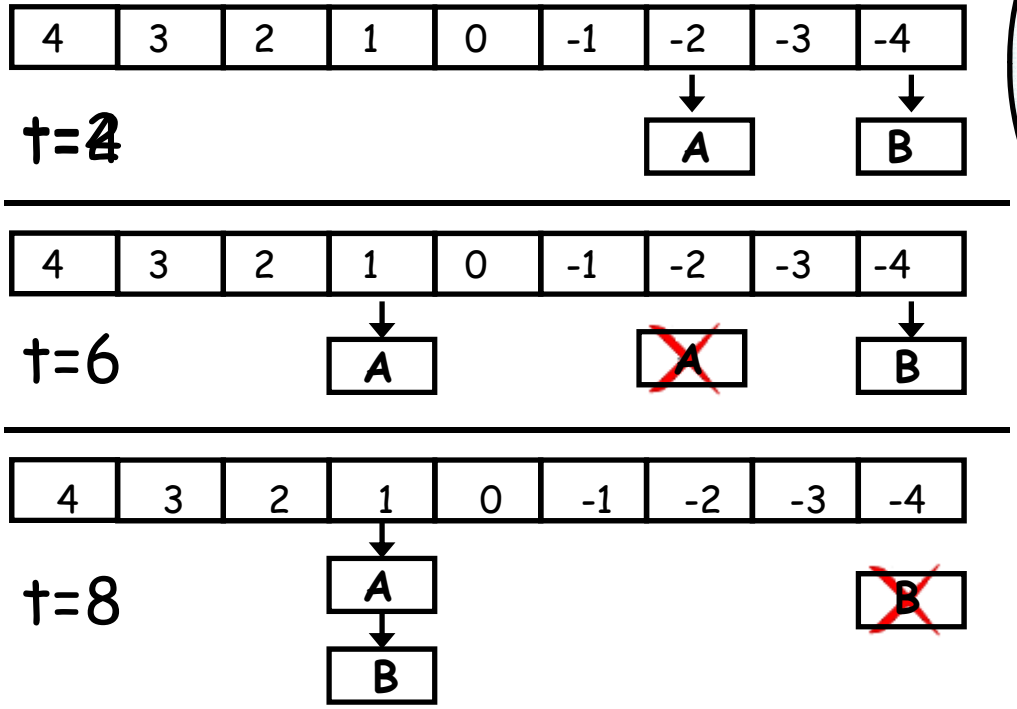
- Observations:
  - The need of peers is guided by playback point
  - Peers play sequentially between jumps
- Smart tracker meshes together peers that have content to exchange
  - Smart Neighbor Selection (SNS) returns list of peers at the same playback point
  - History Neighbor Selection (HNS) returns list of peers that contain needed segments

Architecture - Tracker (2)

Smart Neighborhood Selection (SNS)



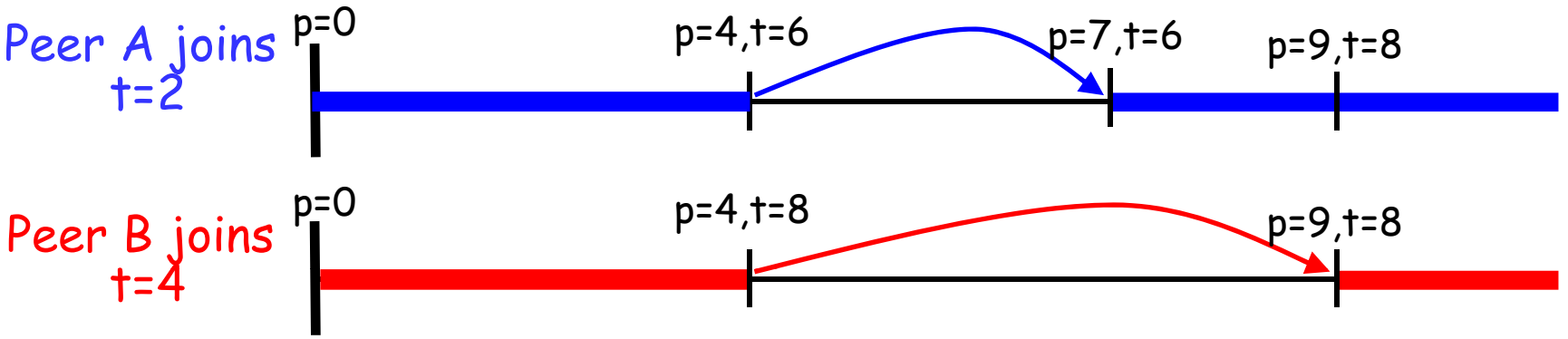




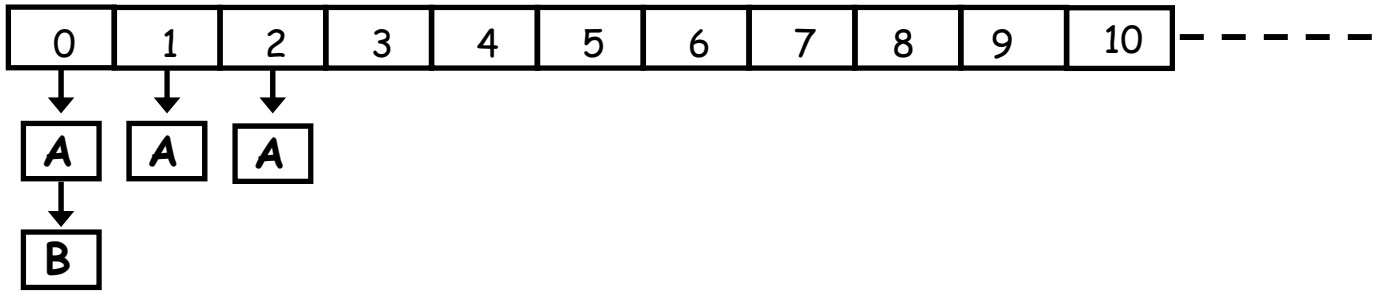
$$\frac{p-t}{c}$$

Architecture - Tracker (3)

History Neighborhood Selection (HNS)

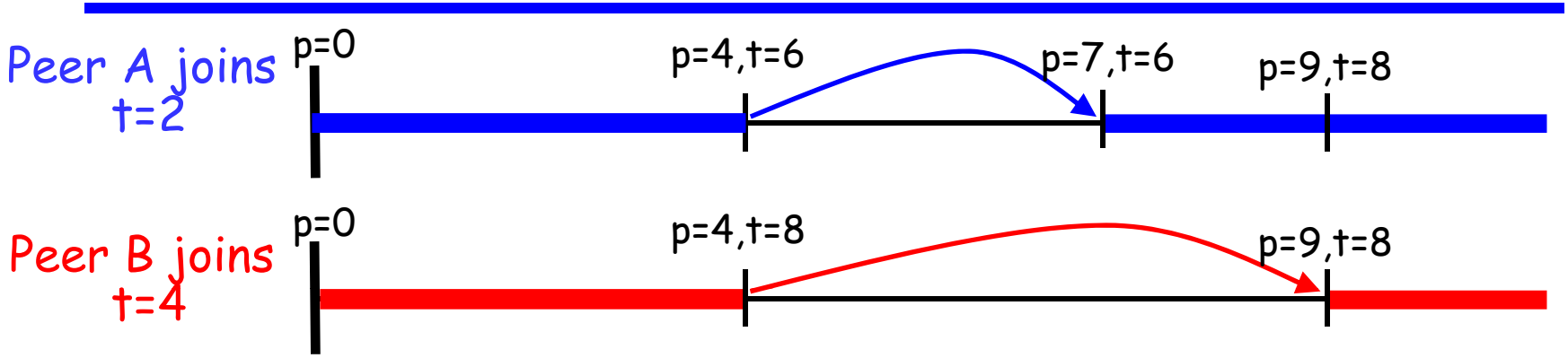


Wall time t=4

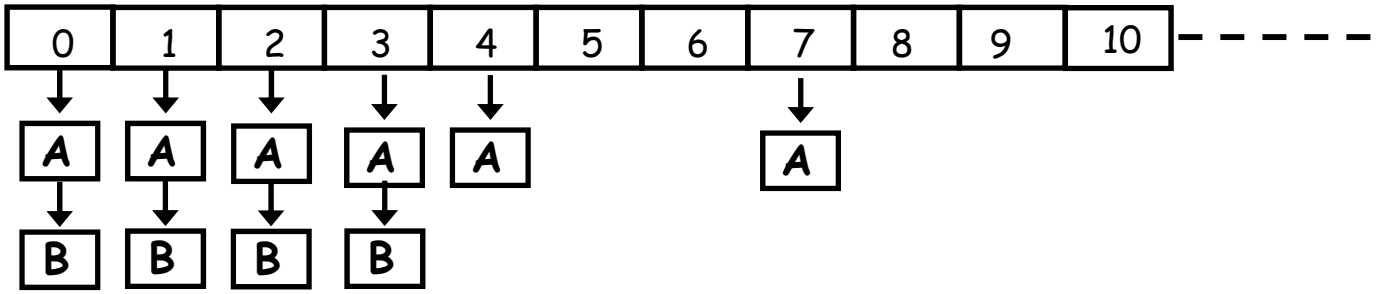


Architecture - Tracker (4)

History Neighborhood Selection (HNS)



Wall time t=7

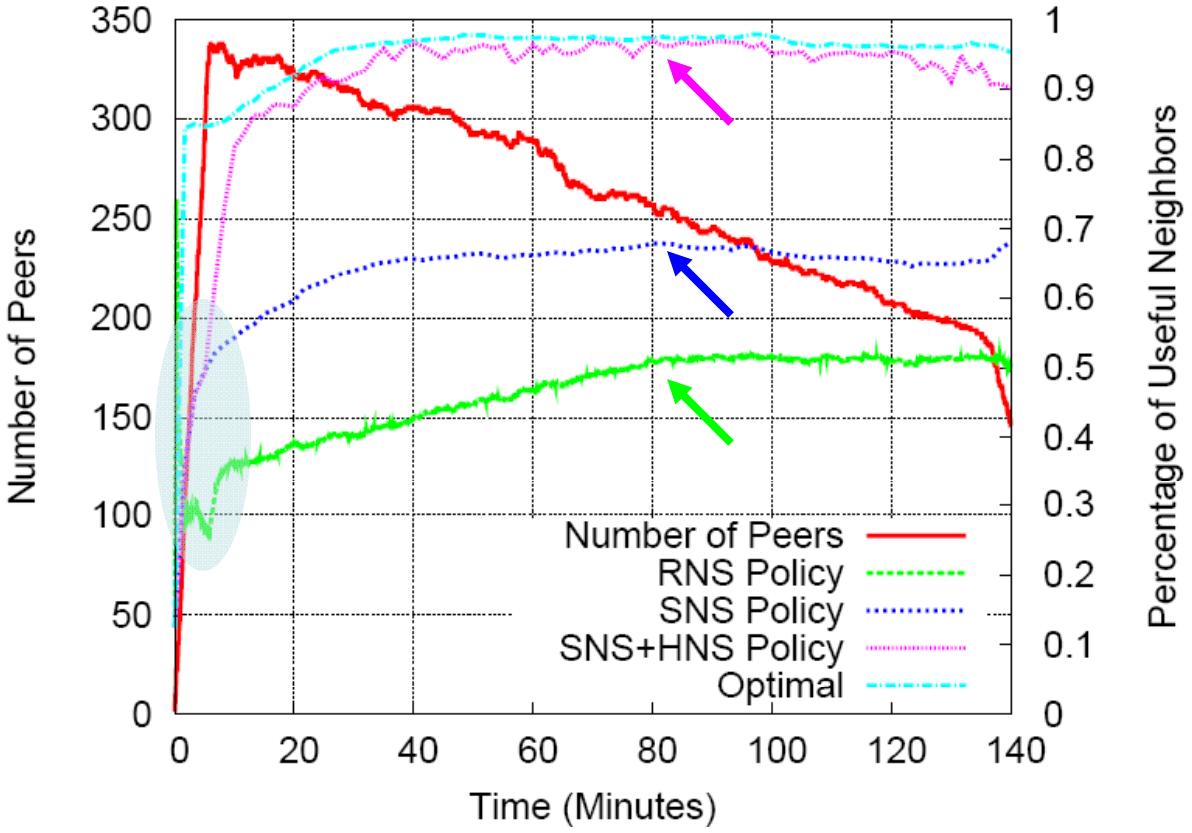


Experimental evaluation

- Experiment setup
  - peers rate limited at 1.5Mbps
  - playback rate of 1Mbps.
  - peer neighborhood size: 10-15
  - 5 active download/upload connections
  - network emulated with a Modelnet cluster of 10 machines connected in a local Gigabit LAN
  - user behavior emulated from real traces collected from a live commercial IPTV service
- Performance metrics
  - jump delay
  - seeder upload bandwidth

Tracker performance

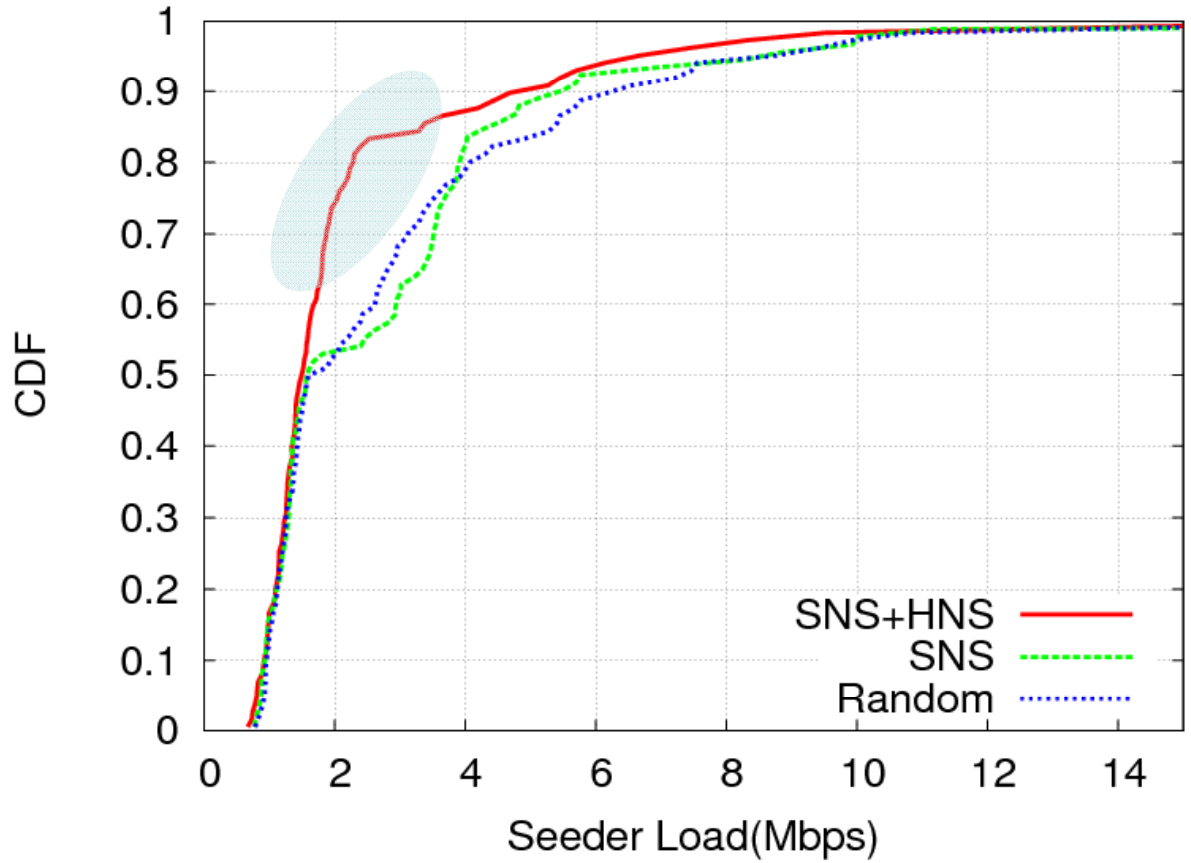
Simulations



- 350 sessions from the most popular video
- peer arrival a Poisson process with λ=1 peers/sec

Tracker performance

System experiments



- Comparisons with Random Neighbor Selection:
  - 3.5% reduction of segments uploaded at the seeder with SNS
  - 22% reduction of segments uploaded at the seeder with SNS+HNS

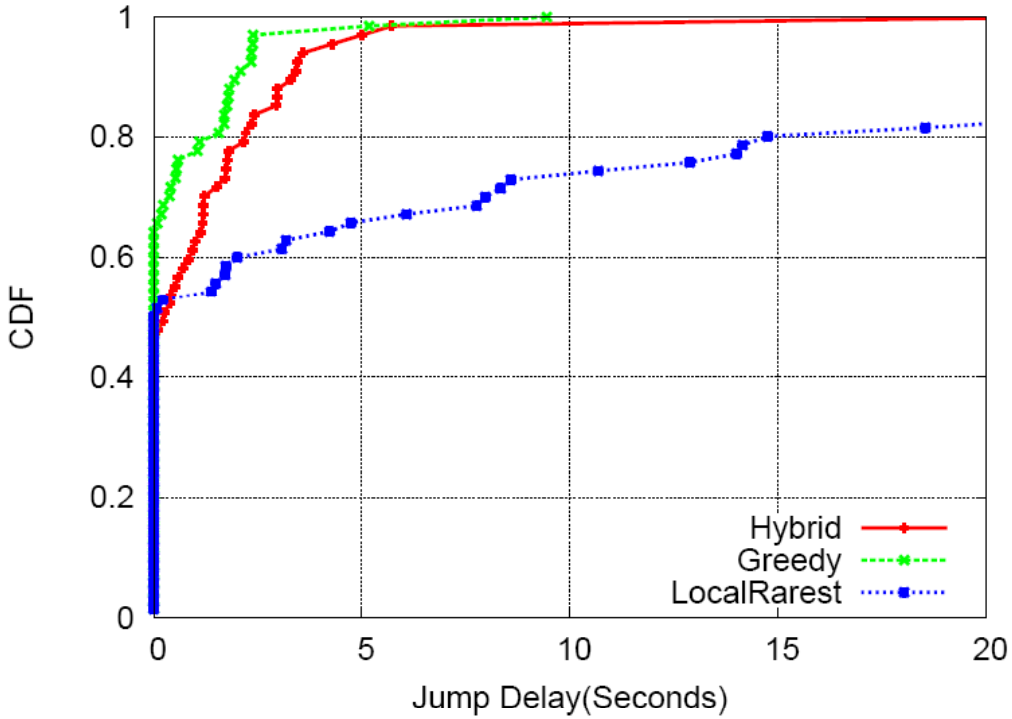
Tracker Scalability

- Tracker scalability depends on number of requests received by tracker:
  - every jump operation
  - triggered by neighborhood health evaluation

- After experimental tests
  - we observed responses < 0.1ms for user behavior of 16,000 "dumb" peers
  - evaluated the trade-off between tracker response time and good topology connectivity

Segment Scheduler performance

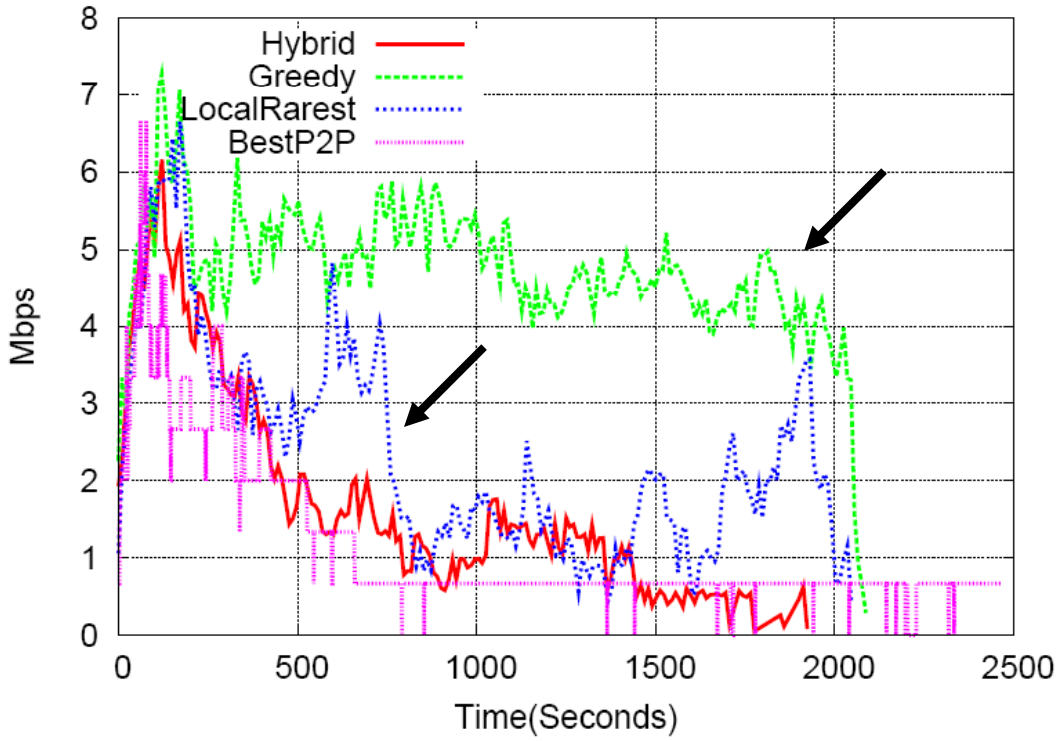
Jump Delay



- Greedy policy achieves the lowest delay
- Adaptive hybrid allocates at least some bandwidth to download rare segments

Segment scheduler performance

Seeder Load



- Proposed adaptive hybrid best compromise between low seeder load and jump delay

Conclusion

- Designed a VoD system that provides good user experience without over-provisioning
- Key mechanisms for this simple design
  - a smart scalable tracker
  - an adaptive hybrid segment scheduler
  - "least useful" peer selection
- System tested with real users during 2008 Olympic games.

Future work

- Extend Kangaroo to support adaptive video quality

- Plans to deploy Kangaroo as a Content Distribution Network.

Questions?