

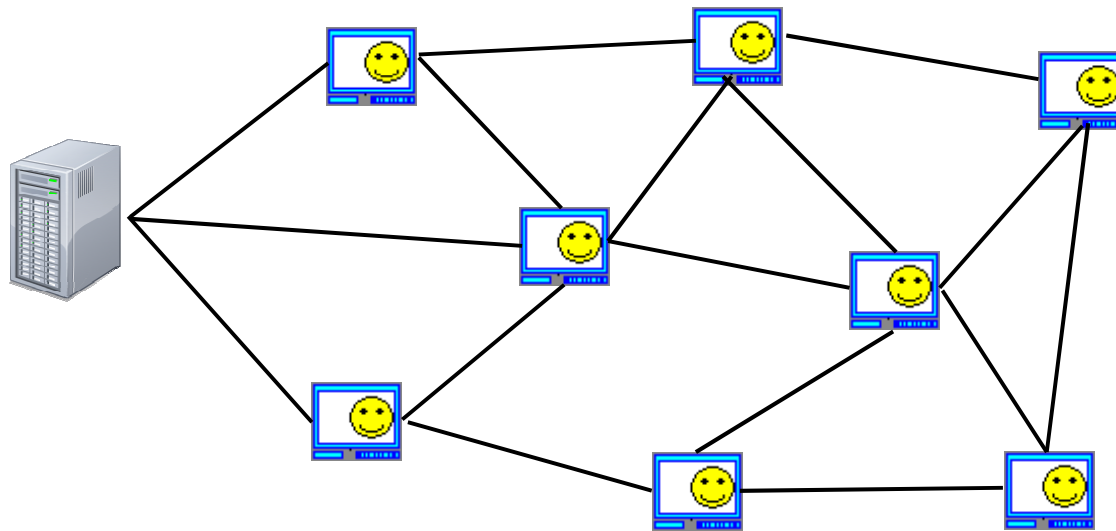
Providing Statistically Guaranteed Streaming Quality for Peer-to-Peer Live Streaming

Lisong Xu, joint work with Miao Wang, and Byrav Ramamurthy
University of Nebraska - Lincoln



P2P Streaming Quality

- ▶ What is a common problem of all P2P streaming applications?
- ▶ All of them currently provide only best-effort streaming quality, and thus we are interested in how to provide guaranteed streaming quality.



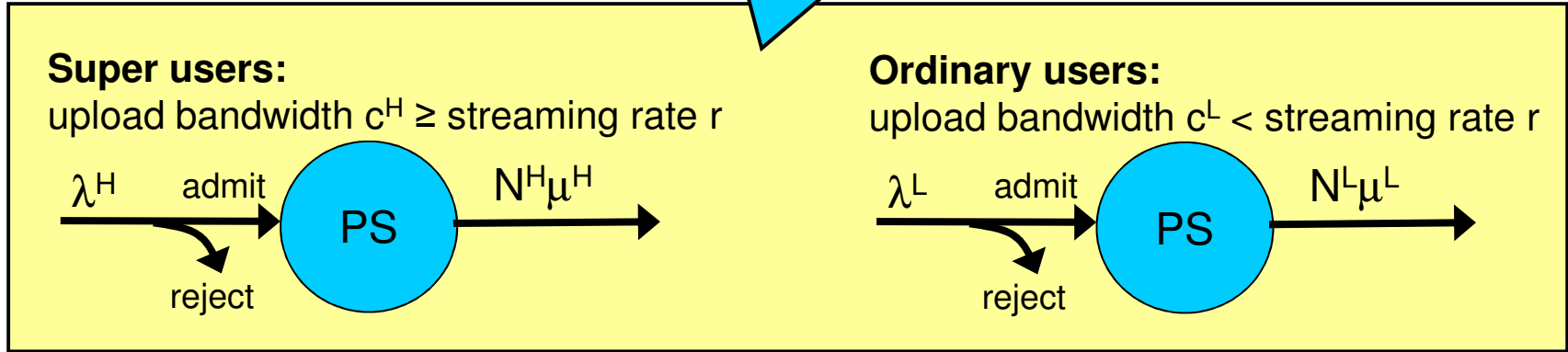
Guaranteed Streaming Quality

- ▶ Hard to achieve guaranteed streaming quality
 - ▶ Reason 1: User Dynamics
 - ▶ Impossible to provide absolute guarantee, and only possible to provide statistical guarantee.
 - ▶ Reason 2: Complicated System
 - ▶ Streaming quality depends on many factors, such as the overall upload bandwidth, overlay topology, block scheduling algorithm, and others.
- ▶ As the first step, our paper studies the following problem
 - ▶ **How to use admission control to provide statistical guarantee of the overall upload bandwidth of a P2P streaming system?**

Queueing model for admission control

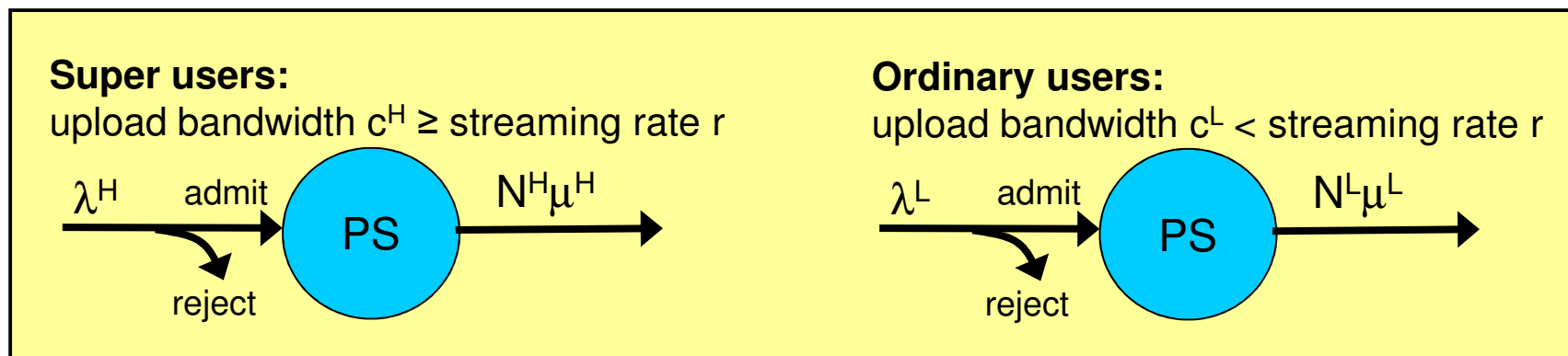
- ▶ 1. Heterogeneous upload bandwidth
 - ▶ Super users have high upload bandwidth.
 - ▶ Ordinary users have low upload bandwidth.
- ▶ 2. User dynamics
 - ▶ New users arrive randomly
 - ▶ Existing users leave randomly
- ▶ 3. Admission control
 - ▶ A new user may be admitted or rejected

System state (N^H, N^L)
 N^H : Current number of super users.
 N^L : Current number of ordinary users.



Formal definition of statistical guarantee

- ▶ We say that a P2P streaming system has sufficient upload bandwidth if the bandwidth demand \leq bandwidth supply.
 - ▶ Bandwidth demand is the total required upload bandwidth, and it is $(N^H+N^L) \times r \times \varepsilon$, where $\varepsilon > 1$ indicates control overhead.
 - ▶ Bandwidth supply is the overall system upload bandwidth, and it is $N^H c^H + N^L c^L + c^S$, where c^S is the streaming server bandwidth.
- ▶ **Statistical Guarantee Problem:** whether a new user is admitted or rejected in order to guarantee that $\text{Prob}[\text{demand} \leq \text{supply}]$ is greater than a threshold.



Admission Control Algorithms

▶ 1. Static Admission Control

Not depend on system state

- ▶ Admits all super users. Randomly admits an ordinary user with a certain probability.

▶ 2. Semi-Static Admission Control

Depend on partial system state

- ▶ Admits all super users. Admits an ordinary user only if the modified demand is less than a certain fraction of the modified supply.
- ▶ Mod. Demand = $(E[N^H] + N^L) \times r \times \epsilon$, Mod. Supply = $E[N^H]c^H + N^Lc^L + c^S$

▶ 3. Dynamic Admission Control

Depend on complete system state

- ▶ Admits all super users. Admits an ordinary user only if the current demand is less than a certain fraction of the current supply.
- ▶ Demand = $(N^H + N^L) \times r \times \epsilon$, Supply = $N^Hc^H + N^Lc^L + c^S$

Performance Metrics

- ▶ We compare these three admission control algorithms according to the following four performance metrics.

- ▶ 1. Algorithm Implementation Difficulty

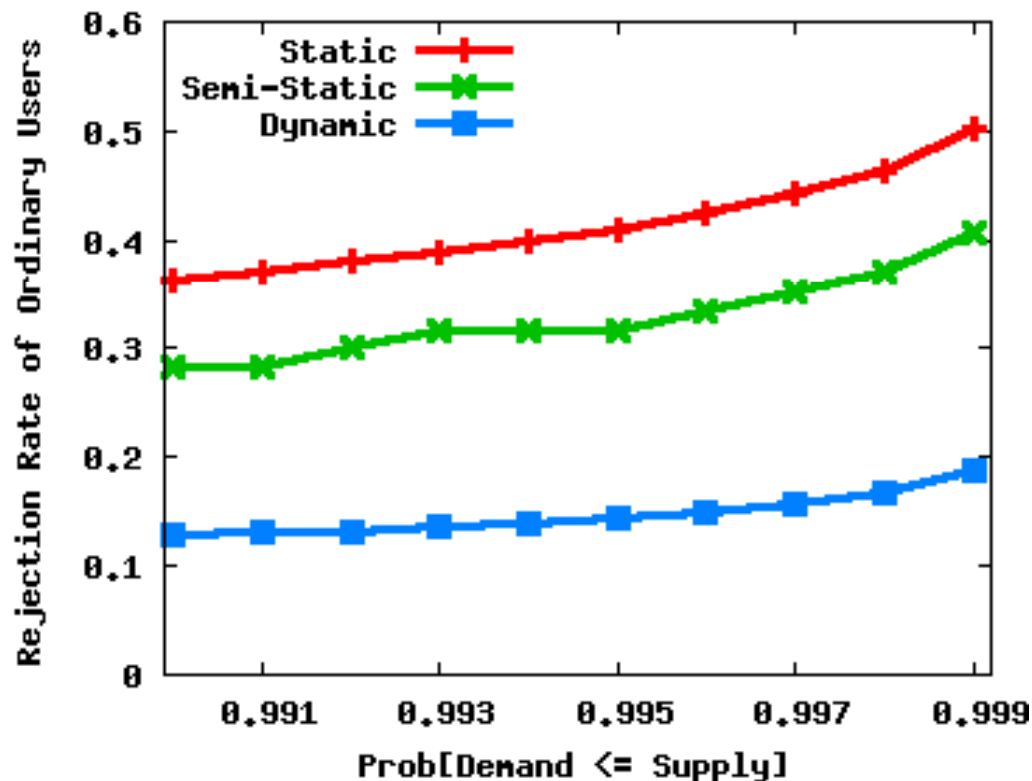
- ▶ 2. User Rejection Rate

- ▶ 3. User Retry Robustness

- ▶ 4. User Behavior Insensitivity

User Rejection Rate

- ▶ To guarantee a certain $\text{Prob}[\text{demand} \leq \text{supply}]$, what percentage of ordinary users should be rejected?



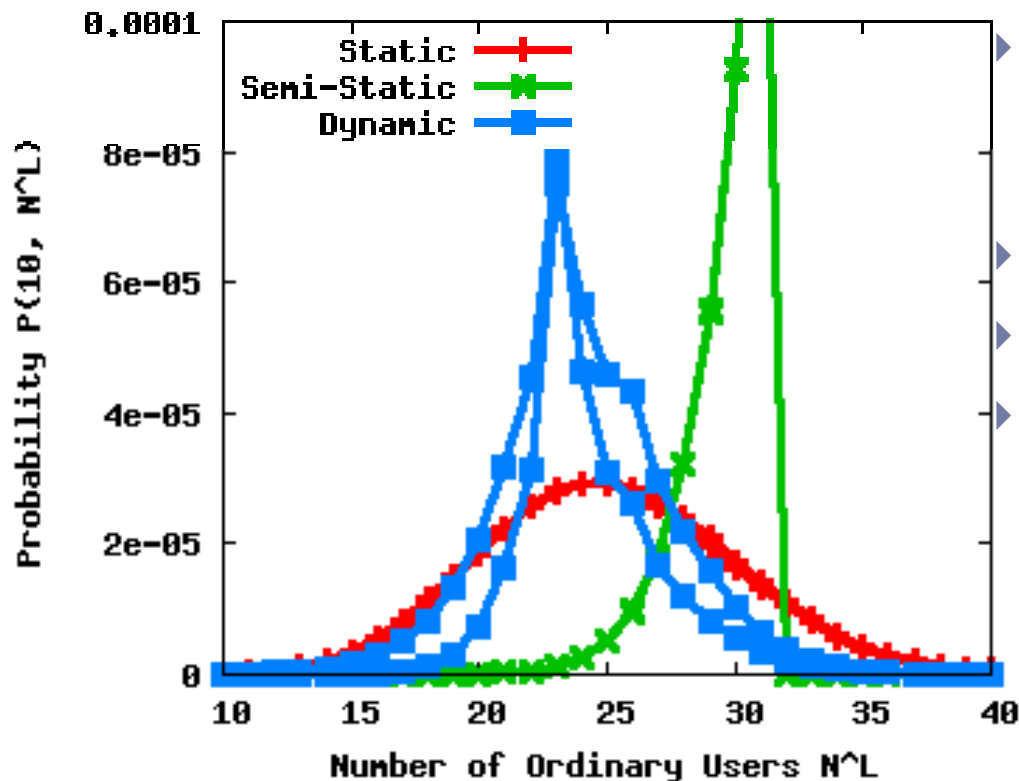
- ▶ Poisson arrivals
- ▶ Exponential lifetime distributions
- ▶ Observations:
 - ▶ 1. For the same algorithm: a higher guarantee probability requires a higher rejection rate.
 - ▶ 2. Rejection: Static (highest)
 - > Semi-Static (medium)
 - > Dynamic (lowest)
 - ▶ 3. Reason: use more state information \Rightarrow achieve a lower rejection rate.

User-Behavior Insensitivity

- ▶ What? Whether an algorithm is insensitive to the fine statistics of user behaviors (i.e., arrival process and lifetime distribution).
- ▶ Why? This is the key to designing an algorithm which can achieve predictable performance in a P2P system with dynamic and changing user behaviors.
- ▶ Definition 1: An algorithm is insensitive to the user lifetime distribution, if the system state distribution $P(N^H, N^L)$ depends on the average but not on the distribution of user lifetimes.
(same average user lifetime \Rightarrow same streaming quality)
- ▶ Definition 2: An algorithm is insensitive to the user arrival process, if the system state distribution $P(N^H, N^L)$ depends on the average but not on the distribution of inter-arrival times.
(same average arrival rate \Rightarrow same streaming quality)

Insensitivity to the Lifetime Distribution

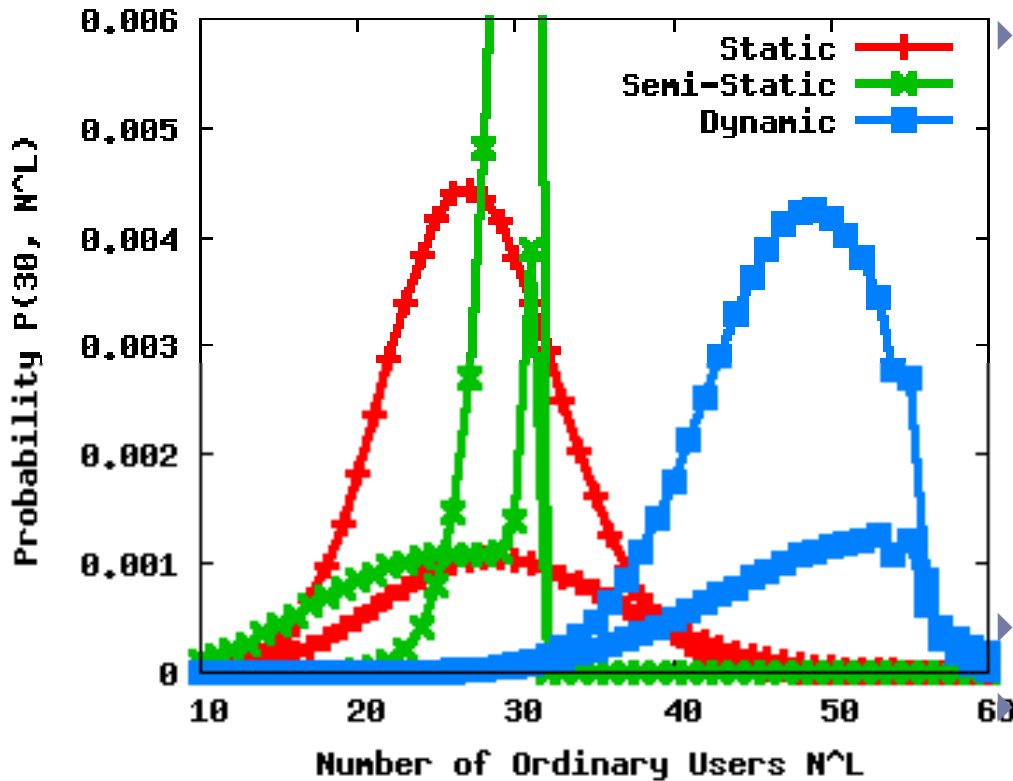
- ▶ Theorem (Proof given in the paper): The sufficient and necessary condition for an algorithm to be insensitive to the user lifetime distribution is that its admission decision does not depend on the current number of super users N^H .



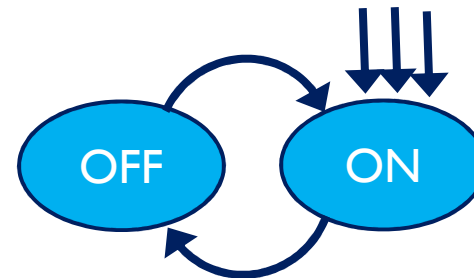
- ▶ Two different Pareto lifetime distributions with the same average lifetime but different shape parameters.
- ▶ Observations:
 - ▶ 1. Static and Semi-Static are insensitive.
 - ▶ 2. Dynamic is sensitive.

Insensitivity to the Arrival Process

- ▶ Unfortunately, we do not have a theorem.



▶ Two different Interrupted Poisson Processes (IPPs) with the same average arrival rate but different OFF durations and different ON durations.

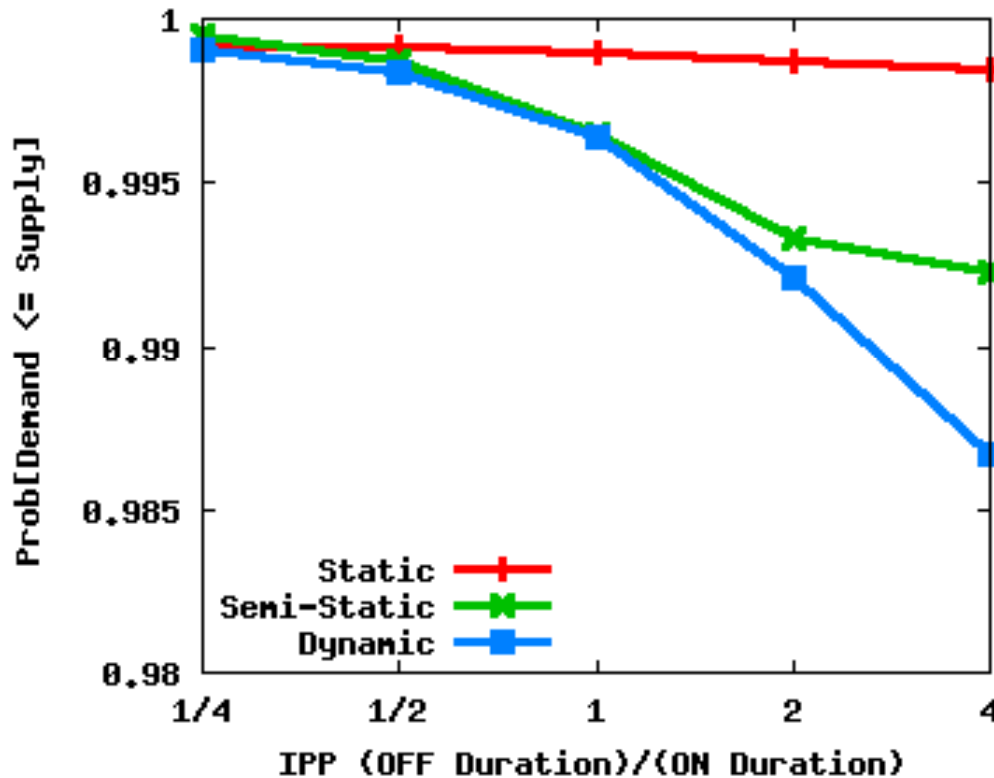


Observations:

- I. All are sensitive.

Insensitivity to the Arrival Process (2)

- ▶ Since all are sensitive, what are their degrees of sensitivity?



▶ Different IPPs with the same average arrival rate but different OFF durations and different ON durations.

▶ Observations:

- ▶ 1. Sensitivity: Static (least sensitive)
< Semi-Static (medium)
< Dynamic (most sensitive)

▶ 2. Reason: use more state information
⇒ become more sensitive

Rejection Rate and Insensitivity

	Used State Information	User Rejection Rate	User-Behavior Insensitivity
Static Algorithm	None	Highest	Least sensitive
Semi-Static Algorithm	Partial	Medium	Medium
Dynamic Algorithm	Complete	Lowest	Most sensitive

The table shows a clear trend: as the amount of state information used increases from 'None' to 'Complete', the user rejection rate decreases from 'Highest' to 'Lowest', and the user-behavior insensitivity increases from 'Least sensitive' to 'Most sensitive'. Blue arrows highlight these trends: a downward arrow in the 'Used State Information' column, an upward arrow in the 'User Rejection Rate' column, and a downward arrow in the 'User-Behavior Insensitivity' column.

Conclusions

- ▶ We studied three admission control algorithms which provide statistical guarantee of the overall system upload bandwidth.
- ▶ There is a fundamental tradeoff between the user rejection rate and user-behavior insensitivity.
- ▶ In the future, we are interested in the admission control for a multiple-channel P2P streaming system: whether to accept or reject
 - ▶ not only the request of a new user to join the system,
 - ▶ but also the request of an existing user to watch a new channel.

Thank You!