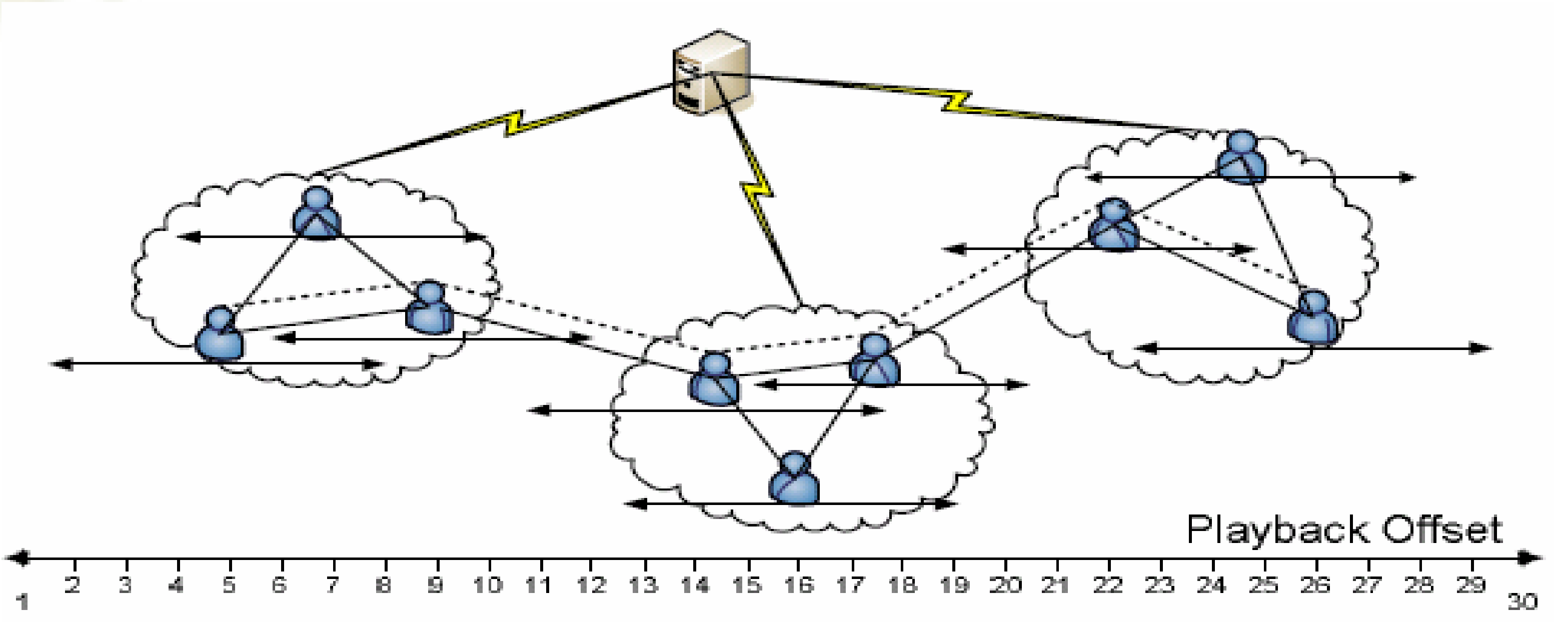
Auction-based incentives and optimal scheduling mechanism design in P2P VoD streaming

Xuanjia Qiu

Sept. 24, 2009

P2P VoD Streaming



Challenges

- Incentive
  - Peers are selfish. They want to download most while upload least.
- Scheduling
  - How to collaborate: who upload what parts of media data to whom at when?
  - Centralized scheduling is impossible.
  - Exacerbated by on-demand properties that lower the levels of content overlap among peers

Possible solutions

- Incentive
  - Tic for Tac
    - successfully applied in file-sharing application, but poorly in VoD application
  - Periodically rebuild the multicast trees
    - increasing the likelihood that a freeloading node's downstream peers will later be upstream of the freeloader and can retaliate by refusing to serve the offender
- Scheduling
  - Rarest first
    - Successfully applied in file-sharing application, but it does not consider the property of streaming
  - Hybrid scheme(place certain weights on rareness and deadline)
    - Can not adapt in a dynamic environment

Our objectives

- Design auction-based mechanism to simultaneously realize two separate goals:
  - Upload incentives
  - Effective block scheduling
- Philosophy:
  - Market can allocate resource optimally
  - Work more, harvest more

Basic concepts

- An **auction** is a process of buying and selling goods or services by offering them up for bid, taking bids, and then selling the item to the highest bidder (s).
- **Bidding** is an offer (often competitive) of setting a price one is willing to pay for something.

P2P VoD Auction Model

- Typical pull-based P2P VoD streaming system
  - Divide each video into many blocks
  - Peers connect to a set of neighbors with similar playback progress
  - Neighbors exchange buffer availability bitmaps (i.e. buffer maps) periodically
  - Each peer maintains a buffer which caches downloaded blocks, that either has been played, or is to be played.

P2P VoD Auction Model (cont.)

- We model media block exchanges among neighboring peers into a collection of decentralized, locally administrated market hosted by each peer.
  - The goods being auctioned and exchanged in the markets are media blocks.
  - In each market, iterated asynchronous auctions take place.
  - In each auction, the host peer plays the role of seller, selling its buffered media blocks to neighbors who bid for the desired ones out of them.
- Idea: modeling media block instead of modeling media flow makes the model more realistic

P2P VoD Auction Model (cont.)

- Each peer is furnished with a budget (some kind of virtual currencies).
  - Winner peers in the auctions gain the rights to, download the wined blocks while pays prices out of its budget to the sellers.
- Idea: Incentive:

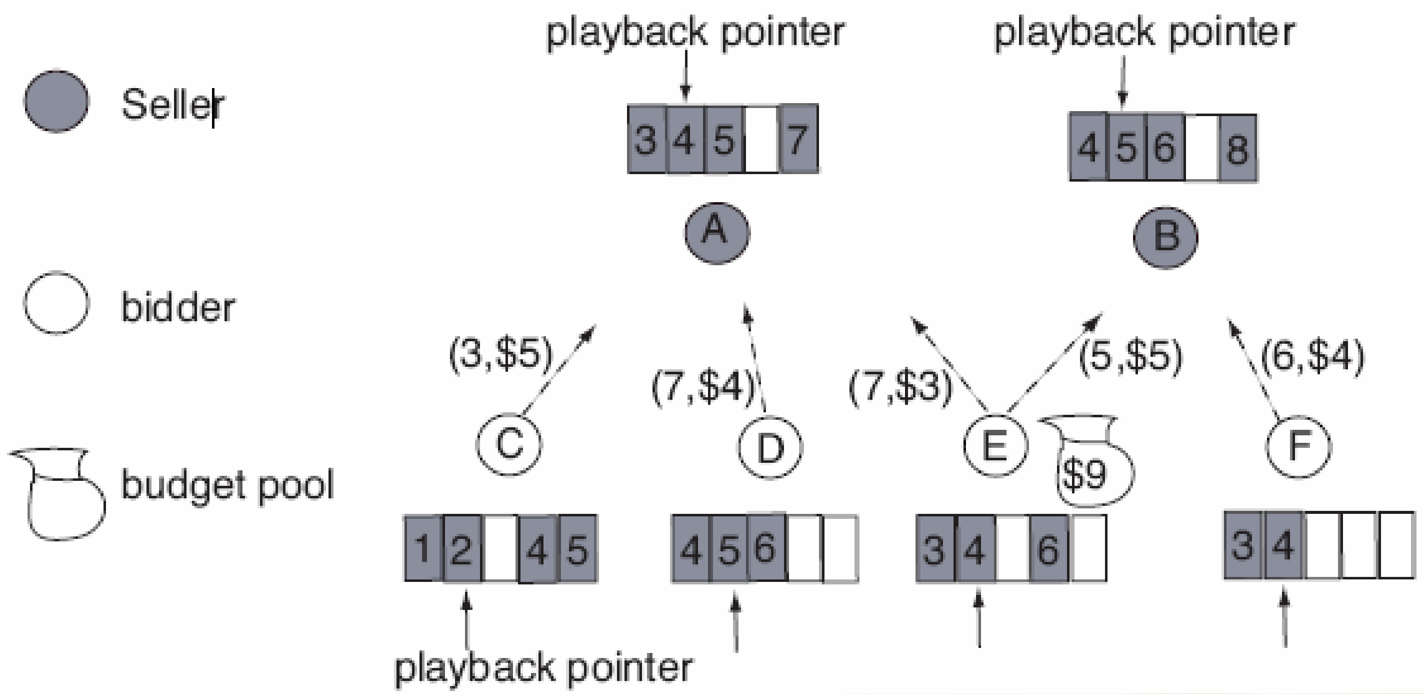
  upload blocks
  -> more budget
  -> more competitive in bidding for blocks
  -> enjoy better viewing experience

P2P VoD Auction Model (cont.)

- Properties:
  - Decentralized
  - Dynamic: with continuously changing blocks and possibly bidders
  - Iterated: along with the streaming process, auctions execute round by round
  - Asynchronous
  - Each peer plays dual roles of both a seller and a bidder.

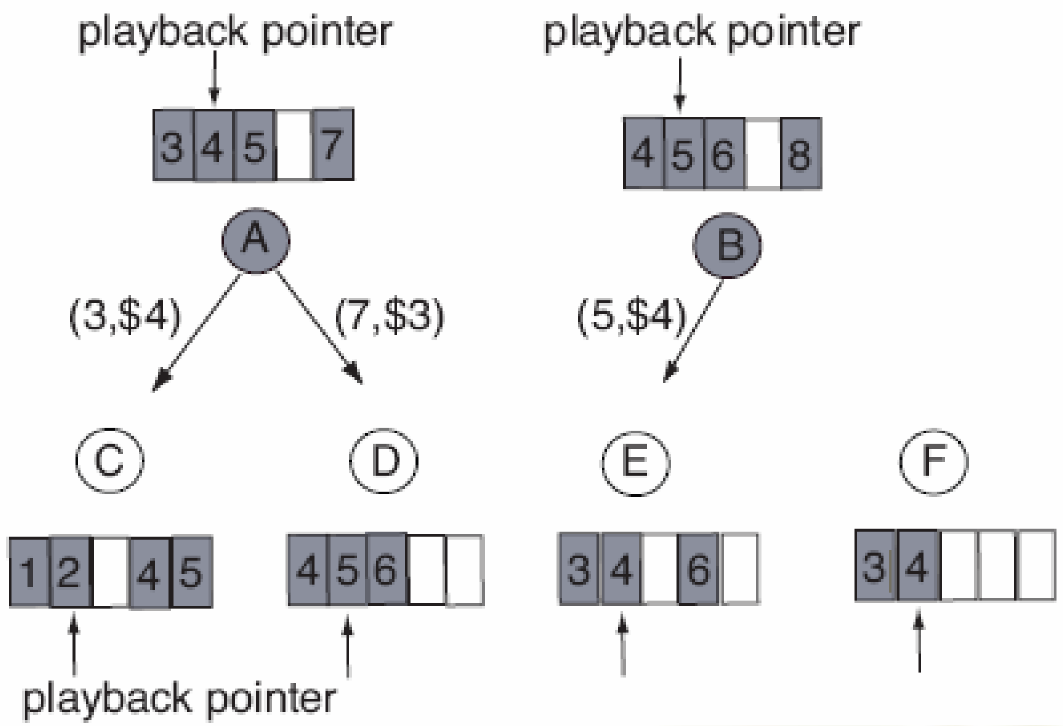
P2P VoD Auction Model (cont.)

- Example: (bidding)



P2P VoD Auction Model (cont.)

- Example: (allocation)



Mechanism design

- For seller
  - A discriminative second price auction with seller reservation
- For bidder
  - A truthful start with iterative price discovery strategy.

Mechanism at seller

- Allocation rule:
  - Sort received bids by bidding prices, and maximally sell blocks in that order, within the available upload bandwidth. (explain with the example picture)

Mechanism at seller

- Charging scheme
  - Select the highest bid-independent charge for each winning bid (indicated by the allocation rule discussed above) as the bidding price from the immediately lower bid.
- Properties:
  - Bid-independent
  - Revenue maximization

Mechanism at seller

- Seller reservation
  - When the number of bids is larger than its maximum upload capacity; otherwise
  - Purpose: keep the market competitive (market price above 0) without revenue loss

$$o_i = \begin{cases} O_i, & \text{if } m > O_i, \\ m-1, & \text{if } m \leq O_i \end{cases}$$

Summary of mechanism at seller

Algorithm 1 Protocol at Seller i (in every interval T)

(a) Allocation

receive bids $\mathbf{b_i}$ from neighbors in $\mathcal{D}_i$
order $\mathbf{b_i}$ in non-increasing order of bidding prices into list $\mathbf{l_s}$
set $o_i = O_i$ if $m = |\mathbf{b_i}| > O_i$; otherwise set $o_i = m - 1$
while $o_i > 0$ **do**
    select next bid $b_{ij}^{(k)} = (I_{ij}^{(k)}, p_{ij}^{(k)})$ in list $\mathbf{l_s}$
    let charge $c_{ij}^{(k)}$ be $p_{ij'}^{(k')}$, price in the subsequent bid in $\mathbf{l_s}$
    send charge $c_{ij}^{(k)}$ to bidder j
    start transfer of block $I_{ij}^{(k)}$ to bidder j
    $o_i \leftarrow o_i - 1$
end while

(b) Upon receiving payment from bidder j for block $I_{ij}^{(k)}$

update budget, $e_i \leftarrow e_i + c_{ij}^{(k)}$

Mechanism at bidder

- An optimization problem
- Definition
  - Valuation of a block: a function (over [0,1], differentiable, non-decreasing, and quasi-linear) reflects the urgency level of downloading the block (playback deadline) and rareness level of the block (potential competition and higher resale chance and price)
  - Utility of a block: valuation minus charging price (Notice: the factual charging price is only revealed after the auction is completed)
  - Marginal utility: valuation divided by charging price

Mechanism at bidder

- Pricing mechanism: truthful start with iterative price discovery
  - First round of bidding at neighbor *l*, bidding price = true block valuation
  - Subsequent rounds of bidding, estimate the market price according to the results of last round of bidding, bidding price = min (market price estimate, true block valuation) .
  - How to estimate market price:
    - *If there are successful bids in the last round*
    - *If all bids fail*

Mechanism at bidder

- Bidding strategy:
  - 1. Decide bidding price according to the “Truthful start with iterative price discovery” strategy
  - 2. Resolve an integer program which maximization the overall utility gained under the constraints :
    - (1) budget constraint
    - (2) for an identical block, only request from a neighbor at a round

Mechanism at bidder

Maximize $\quad \sum_{i\in\mathcal{D}_j}\sum_{k\in\mathcal{K}_{ij}}(v_{ij}^{(k)}(x_{ij}^{(k)}) - p_{ij}^{(k)}x_{ij}^{(k)}) \qquad (5)$

Subject to:

$$\begin{cases} \sum_{i\in\mathcal{D}_j}\sum_{k\in\mathcal{K}_{ij}} p_{ij}^{(k)}x_{ij}^{(k)} \le e_j & & (6) \\ \sum_{i\in\mathcal{D}_j} x_{ij}^{(k)} = z_j^k & \forall k \in \mathcal{K}_{ij} & (7) \\ x_{ij}^{(k)}, z_j^k \in \{0,1\} & \forall i \in \mathcal{D}_j, \forall k \in \mathcal{K}_{ij} & (8) \end{cases}$$

Mechanism at bidder

- Simplification:
  - Compute the price the bidder is willing to pay for each block
  - Select blocks with the highest marginal utilities to bid for

Summary of mechanism at bidder

- Algorithm 2

Initialization

set $\ddot{q}_{ij}$ to a MAX value, $\forall i \in \mathcal{D}_j$

Every Interval T

(a) Bidding

for each neighbor $i \in \mathcal{D}_j$ **do**
    exchange buffer map with i and derive $\mathcal{K}_{ij}$
    set $p_{ij}^{(k)} = \min(v_{ij}^{(k)}, \ddot{q}_{ij}), \forall k \in \mathcal{K}_{ij}$
end for
order blocks in $\cup_{i \in \mathcal{D}_j} \mathcal{K}_{ij}$ in non-increasing order of marginal utility $v_{ij}^{(k)}/p_{ij}^{(k)}$ into list $\mathbf{l_b}$ (excluding duplicates)
$p_{ij}^{(k)} \leftarrow$ price of the first block in list $\mathbf{l_b}$
while $e_j \geq p_{ij}^{(k)}$ **do**
    send bid $(I_{ij}^{(k)}, p_{ij}^{(k)})$ to the corresponding seller i
    $p_{ij}^{(k)} \leftarrow$ price of the next block in list $\mathbf{l_b}$
end while

Summary of mechanism at bidder

(b) After Bidding

$p_i^{max} \leftarrow$ highest bidding price sent to neighbor $i, \forall i \in \mathcal{D}_j$
set the lowest charge at i, $c_i^{min} = p_i^{max}, \forall i \in \mathcal{D}_j$
for each charge $c_{ij}^{(k)}$ received from $i, \forall i \in \mathcal{D}_j$, **do**
    deduct e_j by $c_{ij}^{(k)}$ received
    pay $c_{ij}^{(k)}$ to i
    $c_i^{min} \leftarrow \min(c_i^{min}, c_{ij}^{(k)})$
end for
for each neighbor $i \in \mathcal{D}_j$ **do**
    if no bid was successful (no charge received from i) **then**
        $\ddot{q}_{ij} = p_i^{max} + \delta$
    else $\ddot{q}_{ij} = c_i^{min} - \delta$ **end if**
end for

Analysis

- Incentive compatibility
  - In mechanism design, a process is said to be incentive compatible if all of the participants fare best when they truthfully reveal any private information asked for by the mechanism
  - Seller incentive compatibility & bidder incentive compatibility

Seller incentive compatibility

Theorem 1. *The discriminative second price auction with seller reservation in Algorithm 1 is a revenue-maximizing equilibrium mechanism for a VoD seller*

- The discriminative second price auction that we design is bid-independent →
- Truthful auction →
- (according to revelation principle) →

Buyer incentive compatibility

Theorem 2. *In the auction at seller i described in Algorithm 1, for each block $k \in \mathcal{K}_{ij}$, bidding a price equal to the minimum between the block valuation and the market price at i,* i.e., $\min(v_{ij}^{(k)}, \tilde{p}_i)$, *is a dominant strategy for bidder j.*

Proof of scheduling optimization

- First show that a Nash Equilibrium exists at the stable state of VoD streaming.
- Then show that the optimal solution to the distributed local optimization problem carried out through block auctions in Algorithm 1 and 2 can be combined to construct an optimal solution to the global optimization problem.

Proof of scheduling optimization

Theorem 3. *Without peer joins/departures and VCR operations, the following is a Nash equilibrium in the auction defined by Algorithm 1 at seller i: each participating bidder j with $v_{ij}^{(k)} \geq \tilde{p}_i$ bids and pays $\tilde{p}_i$, all other participating bidders bid their true valuations and lose the auction.*

- Proof:
  - 1. Assume $v_{ij}^{(k)} \geq \tilde{p}_i$
  - 2. Assume $v_{ij}^{(k)} < \tilde{p}_i$

Proof of optimality

Theorem 4. *Algorithms 1 and 2 solve (10),* i.e., *achieves social welfare maximization, in a stable P2P VoD overlay.*

❖ Define the global optimization problem:

$$\text{Maximize} \quad \sum_{j \in \mathcal{N}} \sum_{i \in \mathcal{D}_j} \sum_{k \in \mathcal{K}_{ij}} v_{ij}^{(k)}(a_{ij}^{(k)}) \qquad (10)$$

Subject to:

$$\mathcal{P}_{global} \begin{cases} \sum_{i \in \mathcal{D}_j} \sum_{k \in \mathcal{K}_{ij}} p_{ij}^{(k)} a_{ij}^{(k)} \leq \hat{e}_j & \forall j \in \mathcal{N} \\ \sum_{j \in \mathcal{D}_i} \sum_{k \in \mathcal{K}_{ij}} a_{ij}^{(k)} \leq o_i & \forall i \in \mathcal{N} \\ \sum_{i \in \mathcal{D}_j} a_{ij}^{(k)} = z_j^k & \forall j \in \mathcal{N}, \forall k \in \mathcal{K}_{ij} \end{cases}$$

$$a_{ij}^{(k)}, z_j^k \in \{0, 1\}, \forall j \in \mathcal{N}, \forall i \in \mathcal{D}_j, \forall k \in \mathcal{K}_{ij}$$

- Proof procedure
  - (1) Prove the relaxation form of (5) always hat an integral optimal solution, i.e., the integrality gap of (5) is non-existent
  - (2) the KKT conditions of the relaxation of (5), aggregated across all peers, are equivalent to the KKT conditions of the relaxation of (10)

Evaluation

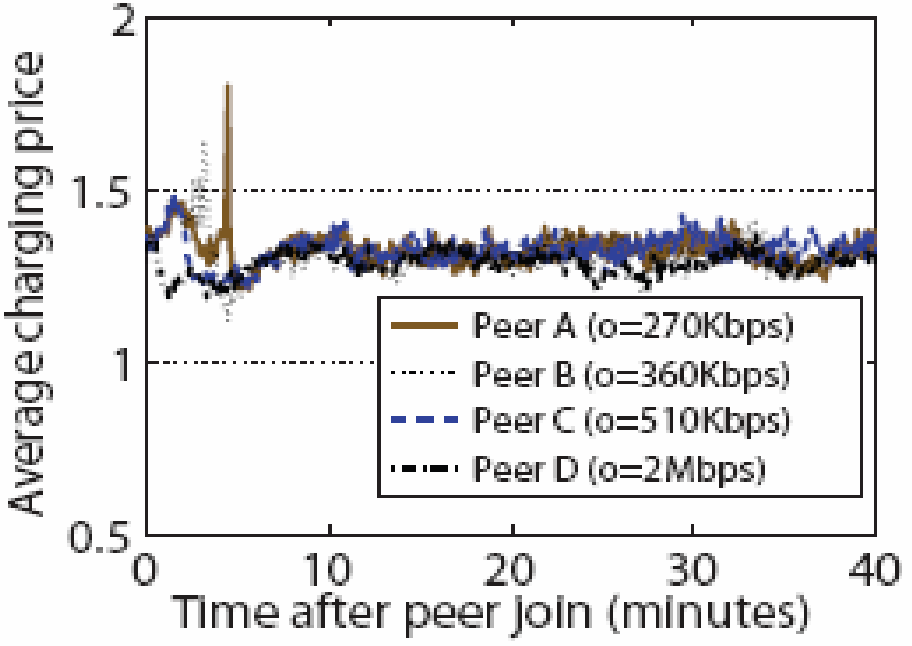
- Multi-thread P2P network simulator in Java
- Supports peer dynamics (VCR operations, peer joins and departures)

Evaluation

- Configuration:
  - 80 minute video is streamed
  - Playback bitrate 450 Kbps
  - Each block = 1/3 seconds of playback
  - Server upload capacity = 10 Mbps
  - Peer upload capacity distribution = Pareto distribution with range=[250 Kbps, 10 Mbps] and k=2 or 3 (default)
  - Peer lifetime = 30 minutes
  - Average duration of VCR operations = 5 minutes
  - Buffer on each peer = 20-minute playback
  - Buffer maps exchange period = 5 sec.

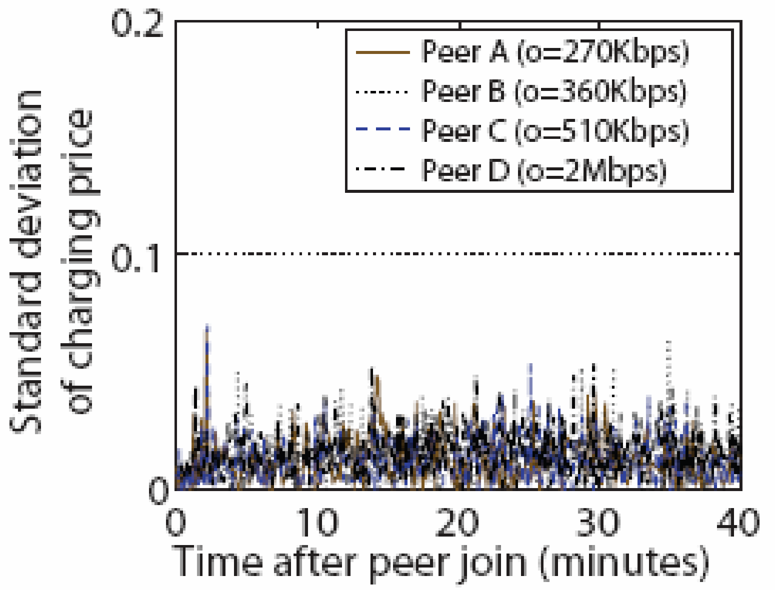
Evaluation

- Stable charging price

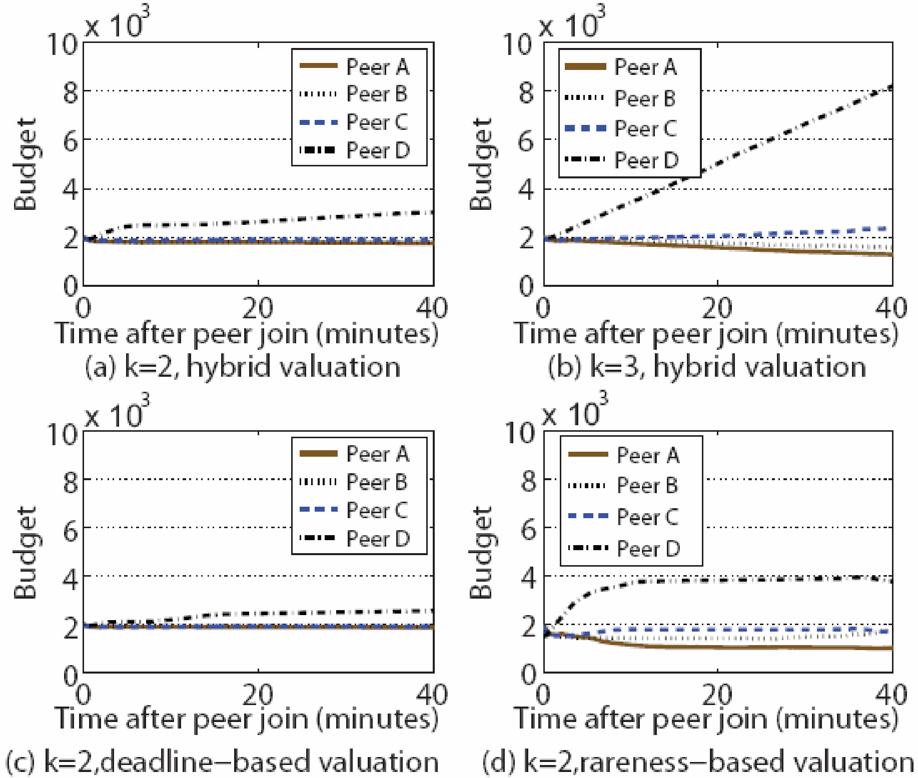


Evaluation

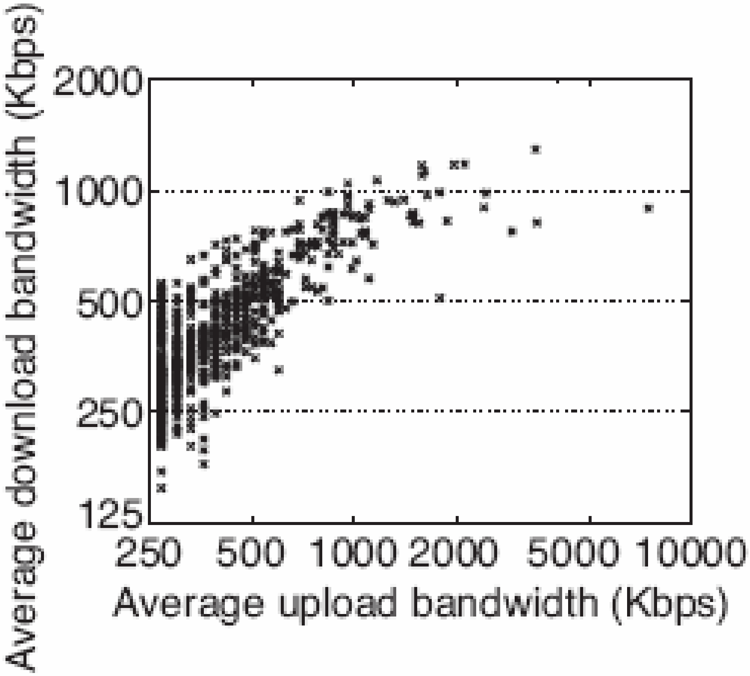
- Stable charging price (cont.)

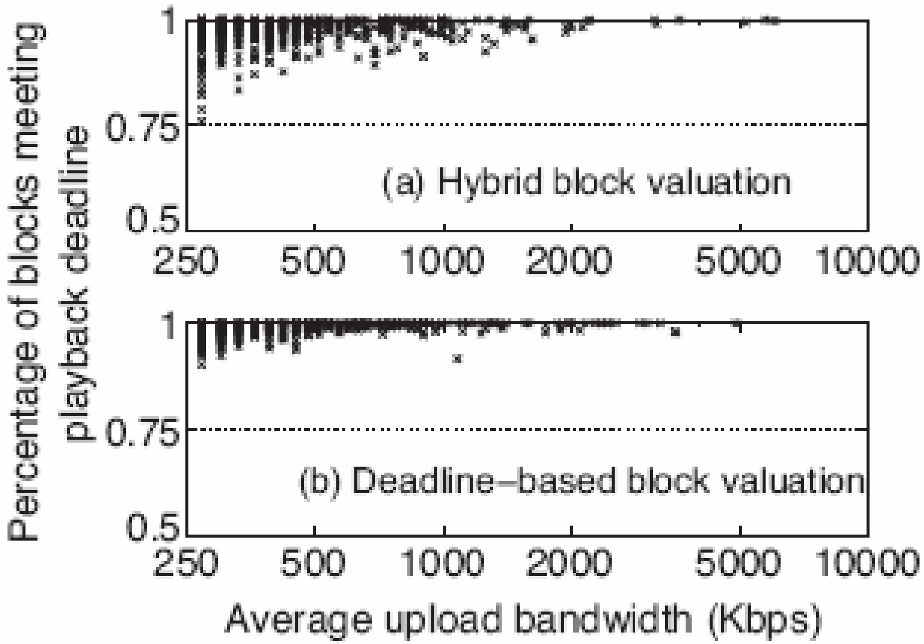


- Evolution of budget at peers with different upload capacities

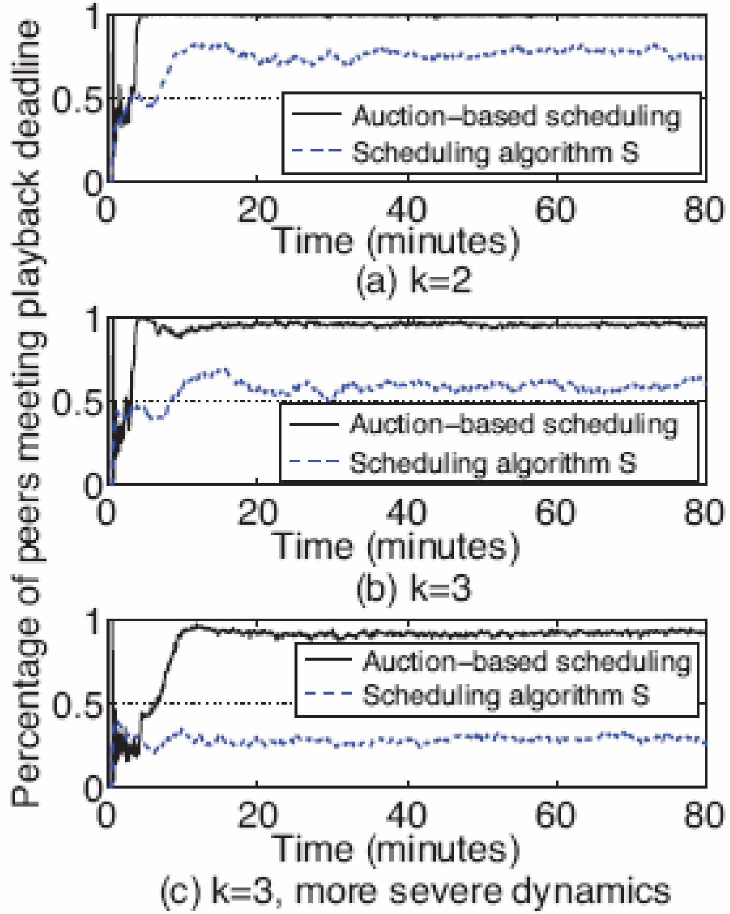


- Incentive

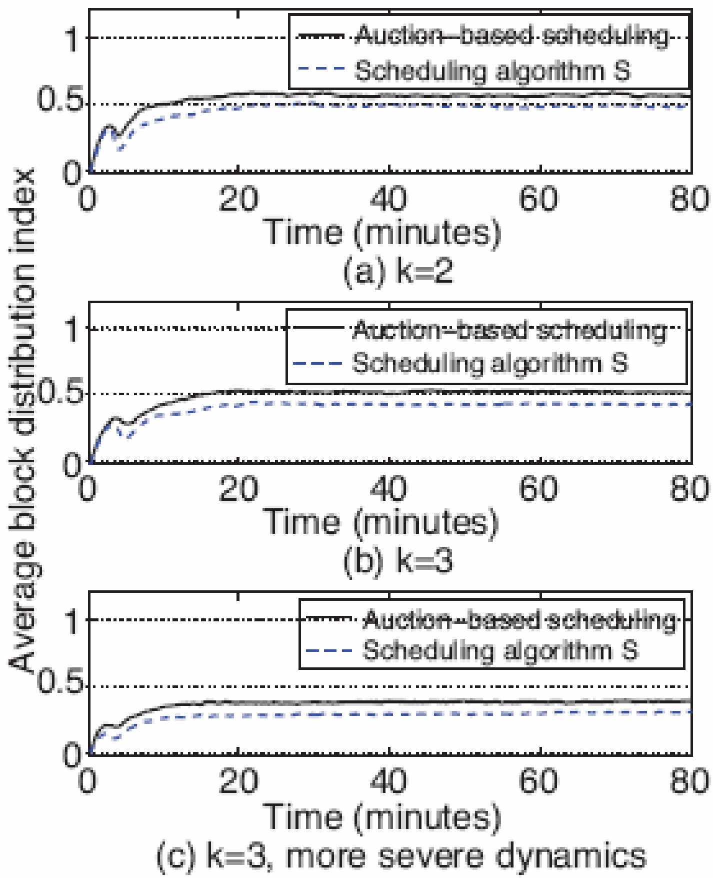

Average download bandwidth (Kbps)
2000
1000
500
250
125
250 500 1000 2000 5000 10000
Average upload bandwidth (Kbps)


Percentage of blocks meeting playback deadline
1
0.75
0.5
(a) Hybrid block valuation
250 500 1000 2000 5000 10000
1
0.75
0.5
(b) Deadline–based block valuation
250 500 1000 2000 5000 10000
Average upload bandwidth (Kbps)

- Performance compared with other mechanisms
  - Evolution of the playback deadline satisfaction



- Performance compared with other mechanisms
  - Evolution of the average block distribution index



Summary

- Design auction-based mechanism
  - to incentive peers to contribute its maximal upload capacity and achieve optimal scheduling of block exchanges among peers
- Analysis proves
  - the incentive compatibility of the scheme
  - the existence of Nash equilibrium under certain conditions
  - at Nash equilibrium, local optimal solutions can be combined to be the global optimal solution.
- Evaluation shows
  - that the scheme exhibits good performance in realistic senarios

Summary

- Chuan Wu, Zongpeng Li, Xuanjia Qiu, Francis C.M. Lau, “Auction-based P2P VoD Streaming: Incentives and Optimal Scheduling”, submitted to INFOCOM 2010
- Future work:
  - Cross-overlay help
  - Research on evolution of budget distribution
  - Wealth condensation? Taxation?
  - Emulation

Question Time