Utility Maximization in Peer-to-Peer Systems

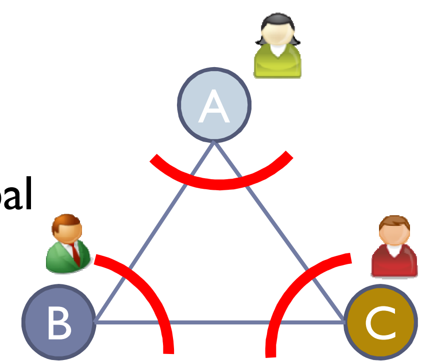
Sudipta Sengupta, Microsoft Research

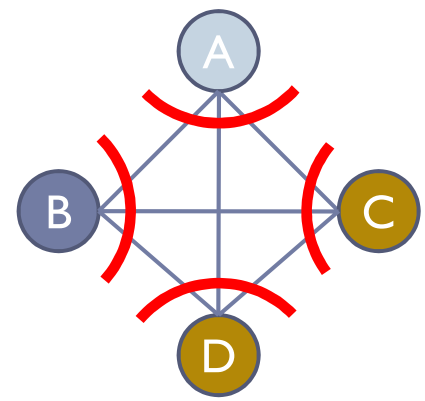
Joint work with M. Chen (CUHK), M. Ponec (Brooklyn Poly),
J. Li, and P. A. Chou (Microsoft Research)

Web Conferencing Application

Multi-party Conferencing Scenario

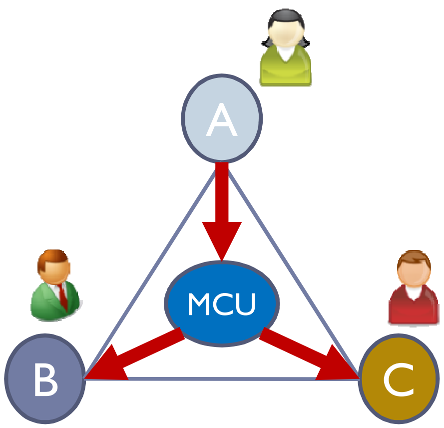
- Every user wants to view audio/video from all other users and is a source of its own audio/video stream
- Maximize Quality-of-Experience (QoE)
- Challenges
  - Network bandwidth limited
  - Require low end-to-end delay
  - Network conditions time-varying
  - Distributed solution not requiring global network knowledge
- Existing Products
  - Apple iChat AV, skype , YAHOO! MESSENGER SightSpeed , hp Halo, cisco TelePresence, Windows Live Messenger , MS Live Meeting





Comparison of Distribution Approaches

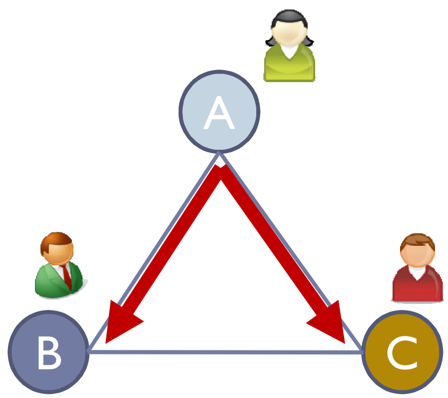
MCU-assisted multicast



High load on MCU, expensive, not scalable with increasing number of peers or groups

Halo

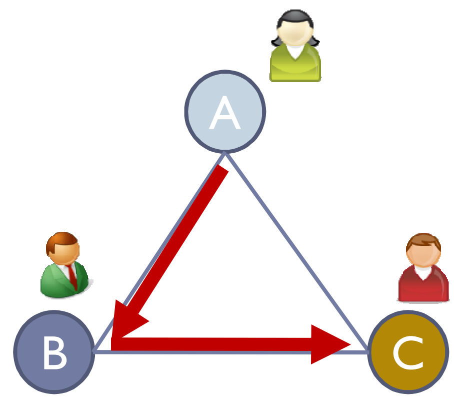
Simulcast



As group size and heterogeneity increases, video quality deteriorates due to peer uplink bandwidth constraint

Apple iChat AV

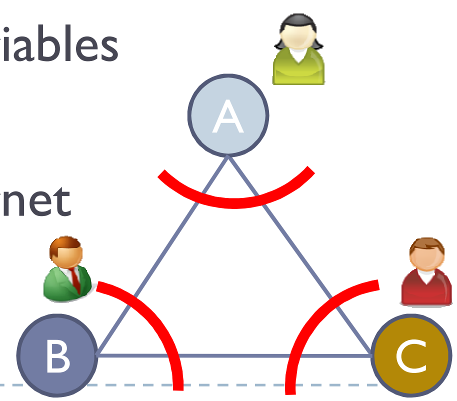
Peer-assisted multicast



Optimal utilization of each peer's uplink bandwidth, no MCU required but can assist as helper

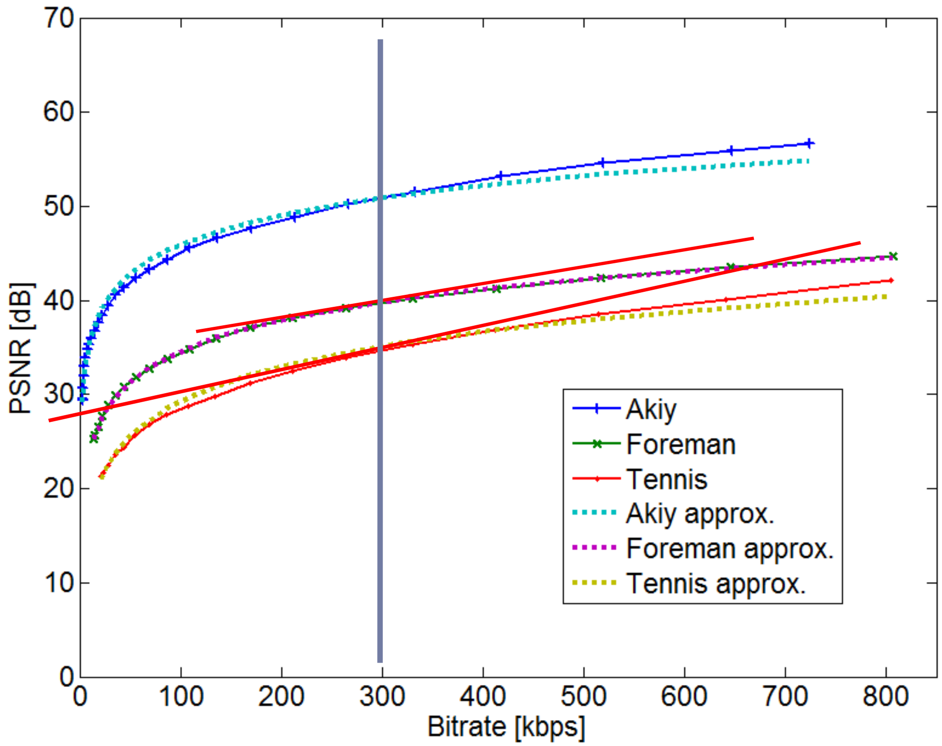
Problem Formulation

- Source s transmitting at rate z_s to all its receivers
- $U_s(z_s)$: (concave) utility associated with video stream of source s
  - Example: PSNR curve
- Only uplinks of peers are bottleneck links
- Maximize total utility of all receivers subject to peer uplink constraints
  - Joint rate allocation and routing problem
  - Linear constraints through introduction of routing variables
  - Concave optimization problem
  - Need distributed solution for deployment in the Internet



Logarithmic Modeling for Utility (PSNR)

- Utility of one peer node defined as $U_s(z_s) = \beta_s \log(z_s)$ **strictly concave**
- Large amount of motion ➔ large β_s
- Peers' utility might change from time to time as they speak/move…



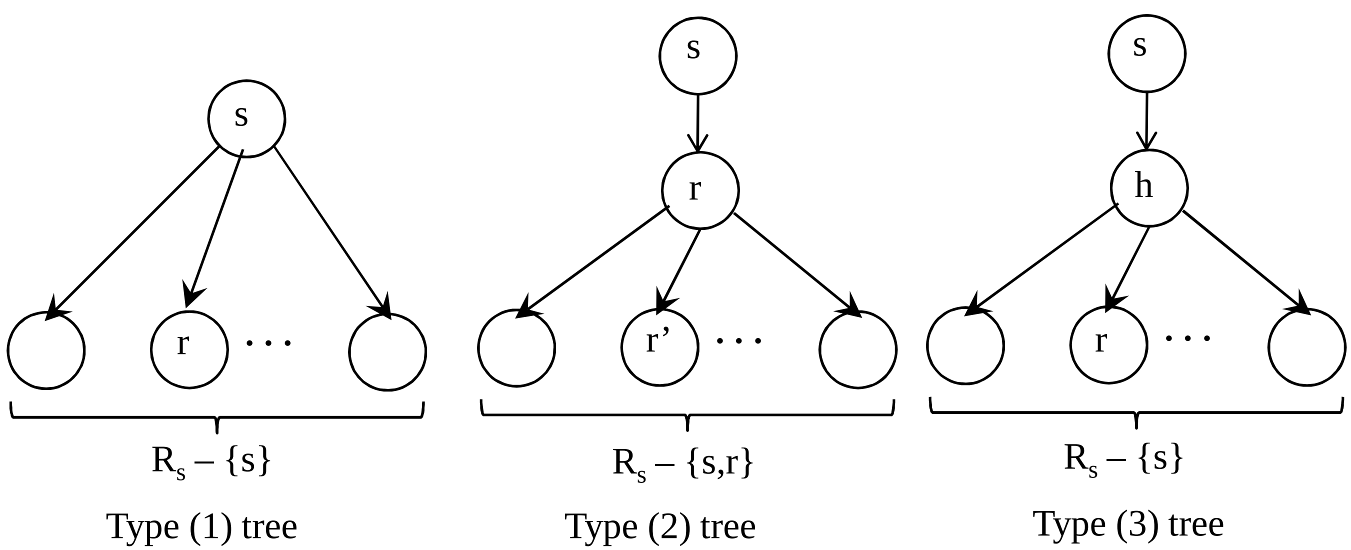
Convex Optimization Problem

$$\max_z \quad \sum_{s \in S} |R_s| U_s(z_s)$$
$$\text{s.t.} \quad \text{the achievable set of } z$$

- S: set of sources
- R_s : set of receivers for source s
- What is the feasible region for rates $\{z_s\}$?
  - Only peer uplink capacities are bottleneck
  - Allow intra-source or inter-source network coding ?

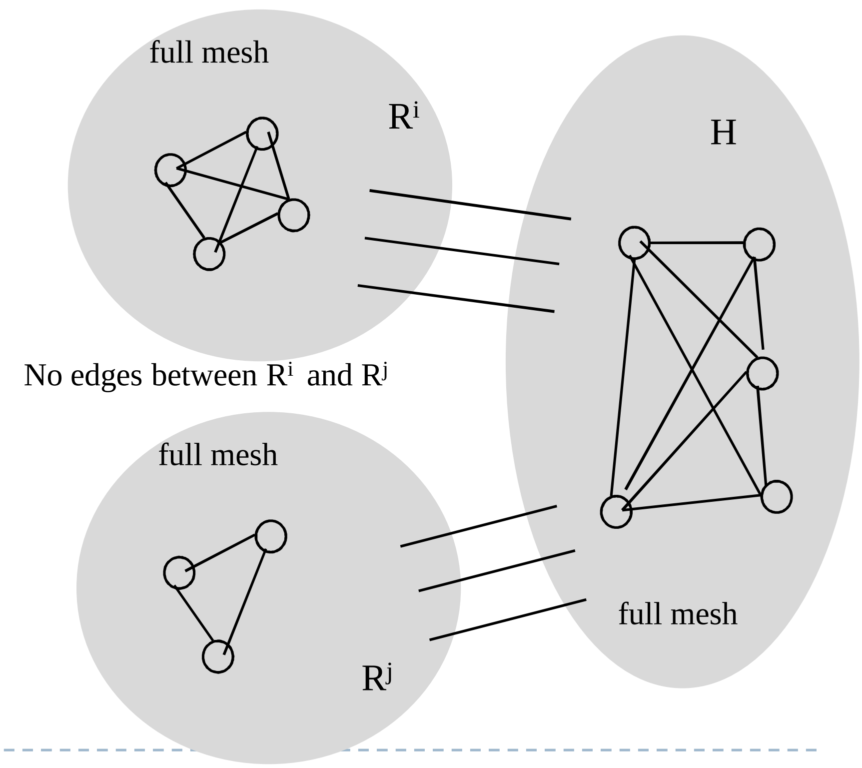
Rate region with Network Coding

- Arbitrary link capacities
  - Routing $\subseteq$ Intra-source coding $\subseteq$ Inter-source coding
- Node uplink capacities only, single source
  - Mutualcast Theorem [Li-Chou-Zhang 05]
  - Routing along linear number of trees achieves min-cut capacity



Rate region with Network Coding …

- Node uplink capacities only, multiple sources
  - No inter-source coding: Linear number of MutualCast trees per source achieve rate region [Sengupta-Chen-Chou-Li 08]
  - Allow inter-source coding: Linear number of MutualCast trees per source achieve rate region [Sengupta-Chen-Chou-Li 08] (some restriction on structure of receiver sets)



New Tree-rate Based Formulation

$$\max_x \quad \sum_{s \in S} |R_s| U_s \left(\sum_{m \in s} x_m \right)$$

$$\text{s.t.} \quad y_j \le C_j, \quad j \in J$$

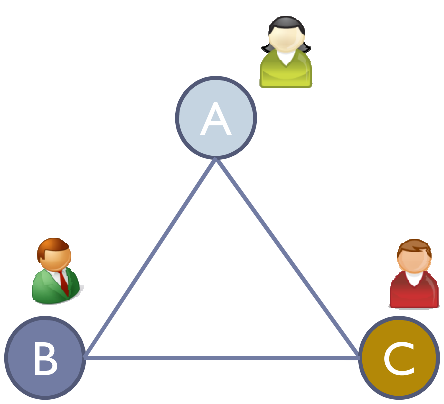
- (Non-strictly) Convex optimization problem with linear constraints
  - y_j : Uplink usage of peer j
  - x_m ($m \in s$): Rate on tree m of source s
  - C_j : Uplink capacity of peer j

Related Work

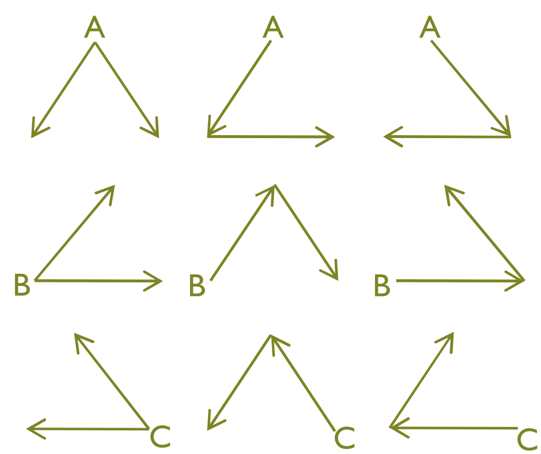
- Utility maximization framework for single-path multicast without network coding [Kelly-Maullo-Tan 98]
- Extensions (without network coding)
  - Multi-path unicast [Han et al 06, Lin-Shroff 06, Voice 06]
  - Single-tree multicast [Kar et al 01]
- Extensions (with single-source network coding)
  - Multicast [Lun et al 06, Wu-Chiang-Kung 06, Chen et al 07]
- This work
  - P2P multicast with multi-source network coding

Need Distributed Rate Control Algorithm

- Best possible rate region achieved by depth-1 and depth-2 trees
  - Determine rate z_s for each source s
  - Determine rates x_m for each source (how much to send on each tree)
- Global knowledge of network conditions or per-source utility functions *should not be required*
  - Adapt to uplink cross-traffic
  - Adapt to changes in utility function (user moving or still)



3 peers



9 multicast trees

Packet Marking Based Primal Algorithm

- Capacity constraint relaxed and added as penalty function to objective

$$\max_{\{x_m\}} \sum_{s\in S} |R_s| U_s(z_s) - \sum_{h\in H} G_h(y_h) - \sum_{j\in J} \int_0^{y_j} q_j(w)\, dw$$

- $q_j(w) = \frac{(w-C_j)^+}{w}$ (packet loss rate or ECN marking probability)
- Simple gradient descent algorithm

$$\dot{x}_m = f_m(x_m)\left(|R_s| U_s'(z_s) - \sum_{h\in m} b_h^m G_h'(y_h) - \sum_{j\in m} b_j^m q_j(y_j) \right)$$

- Global exponential convergence

Queueing Delay Based Primal-Dual Algorithm

- Lagrangian multipliers p_j for each uplink j

$$L(x,p) = \sum_{s \in S} |R_s| U_s(z_s) - \sum_{j \in J} p_j (y_j - C_j)$$

- Primal-dual algorithm

$$\dot{x}_m = k_m \left(U_s'(z_s) - \frac{1}{|R_s|} \sum_{j \in m} b_j^m p_j \right)$$

$$\dot{p}_j = \frac{1}{C_j} (y_j - C_j)_{p_j}^{+},$$

- p_i can be interpreted as queueing delay on peer uplink j
- $\frac{1}{|R_s|} \sum_{j \in m} b_j^m p_j$ can be interpreted as average queueing delay of a branch on tree m

Convergence behavior of Primal-Dual algorithm

- There exist cases where primal-dual system does not converge in multi-path setting [Voice 06]
- Positive Results [Chen-Ponec-Sengupta-Li-Chou 08]
  - For P2P multi-party conferencing, all (x,p) trajectories of the system converge to one of its equilibria if for source s, all its k_m ($m \in s$) take the same value
  - For P2P content dissemination , all (x,p) trajectories of the system converge to one of its equilibria if a mild condition (involving k_m and C_j) is satisfied

Convergence behavior of Primal-Dual algorithm

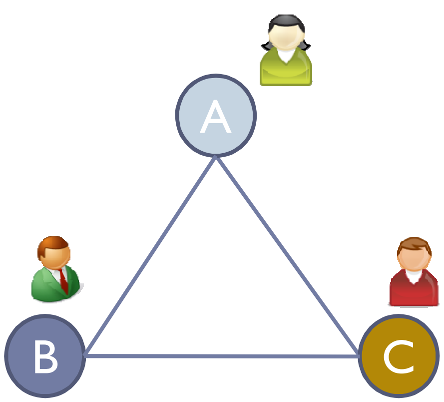
- Trajectories of the system converge to an invariant set, which contains equilibria and limit cycles
  - On the invariant set, the non-linear system reduces to a marginally stable linear system
- Trajectories of the system converge to its equilibria if p is completely observable through [z, y^H] in the reduced linear system
- Mild condition for P2P dissemination scenario
  - *For all* $1 \leq i \neq j \leq n$, $\xi_i \neq \xi_j$, *where*

$$\xi_l = \begin{cases} \frac{(n_l-1)n_l}{C_l} k_{ll}, & 1 \leq l \leq n_s; \\ \frac{1}{C_l} \sum_{j:l \in R_j} (n_j - 1)^2 k_{jl}, & otherwise \end{cases}$$

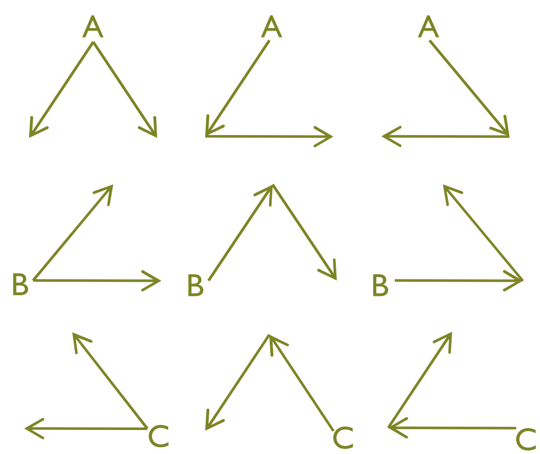
  - $k_{ii} < \frac{C_i}{2C_j} k_{ij}$, *for all* $1 \leq i \leq n_s$ *and* $n_s < j \leq n$.

Implementation of Primal-Dual Algorithm

- What each peer node does?
  - *Sending* its video through trees for which it is a root
  - *Adapting sending rates*
  - *Forwarding* video packets of other peers
  - *Estimating* queuing delay



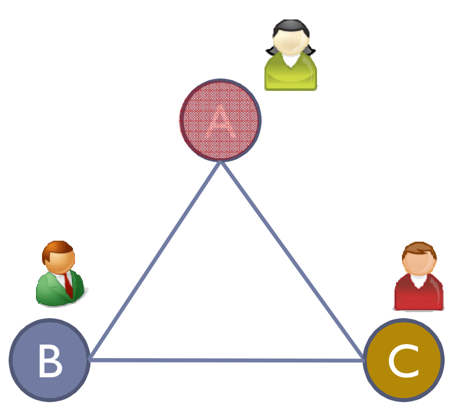
3 peers



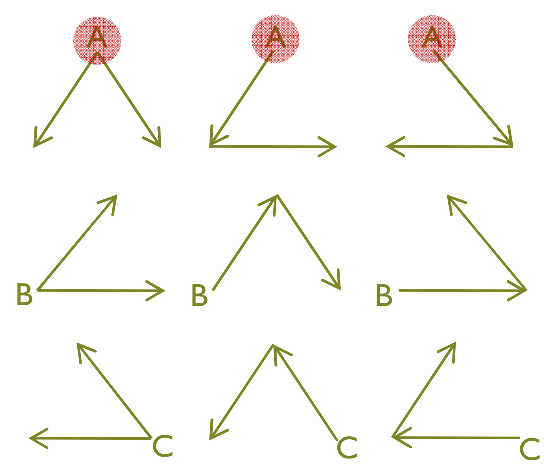
9 multicast trees

Implementation Details

- What each peer node does?
  - *Sending* its video through trees for which it is a root
  - *Adapting sending rates*
  - *Forwarding* video packets of other peers
  - *Estimating* queuing delay



3 peers

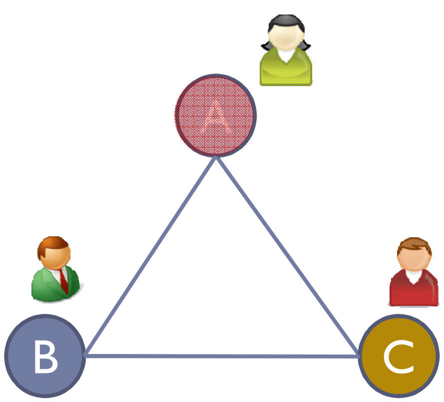


9 multicast trees

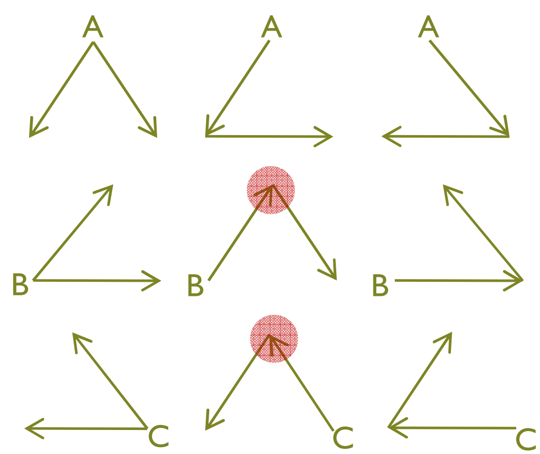
Implementation Details

- What each peer node does?
  - *Sending* its video through trees for which it is a root
  - *Adapting sending rates*
  - *Forwarding* video packets of other peers
  - *Estimating* queuing delay

Helper's functionality

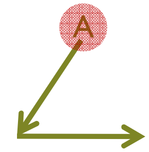


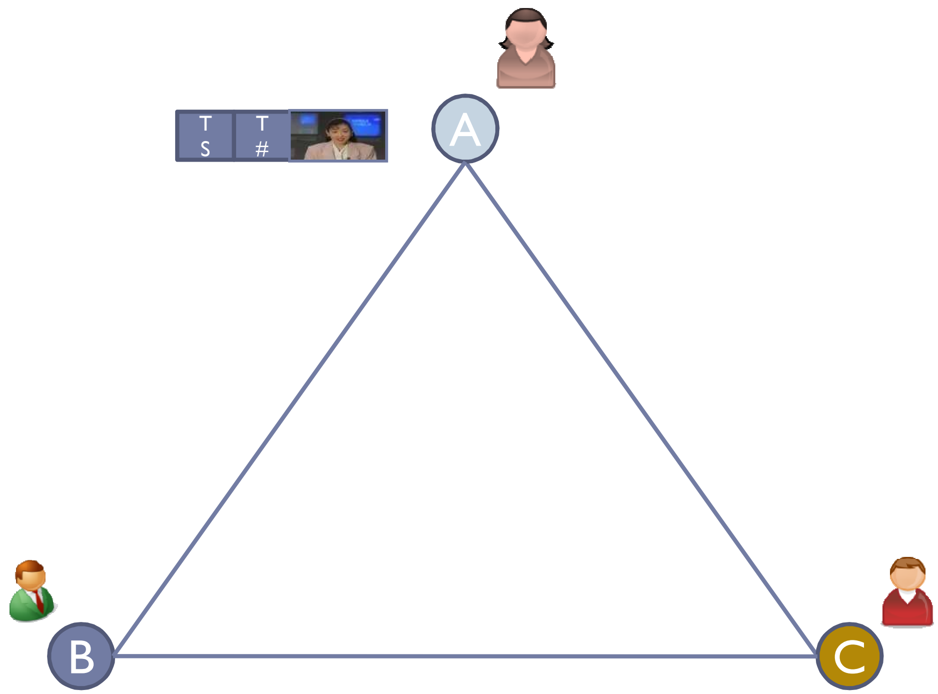
3 peers



9 multicast trees

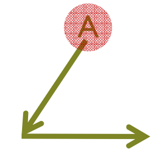
Sending & Forwarding Video

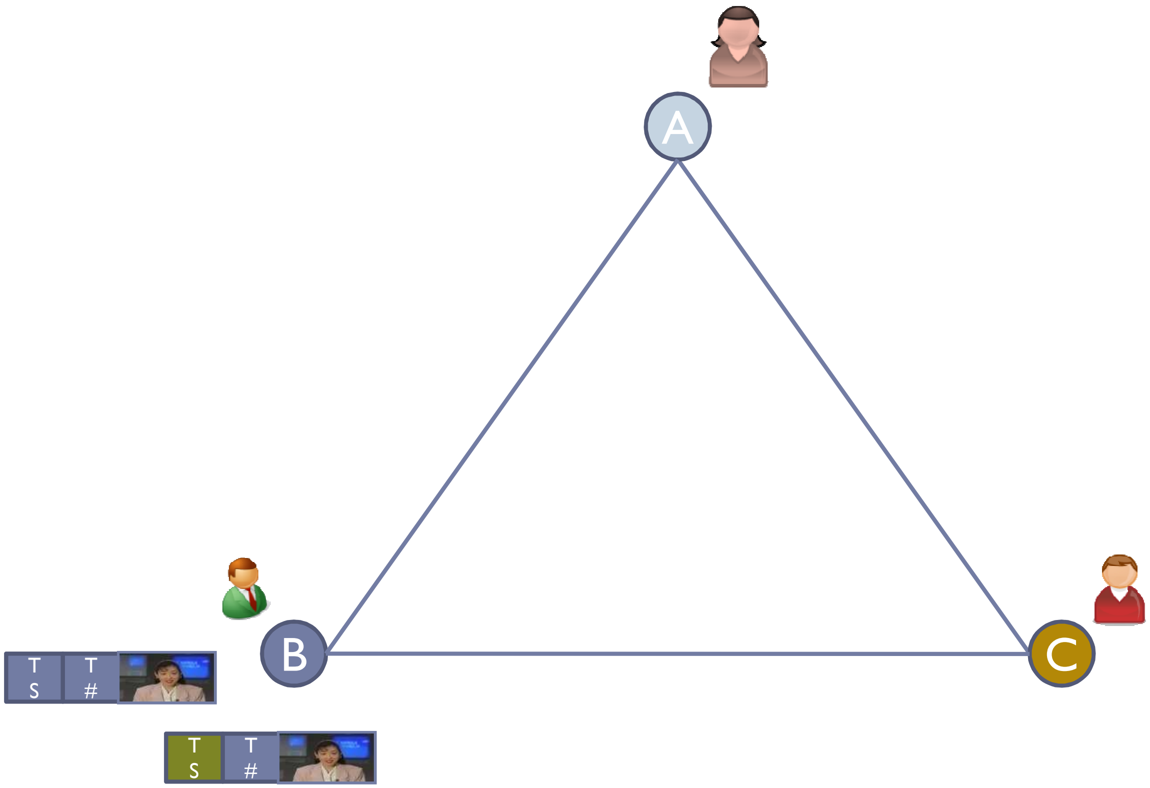




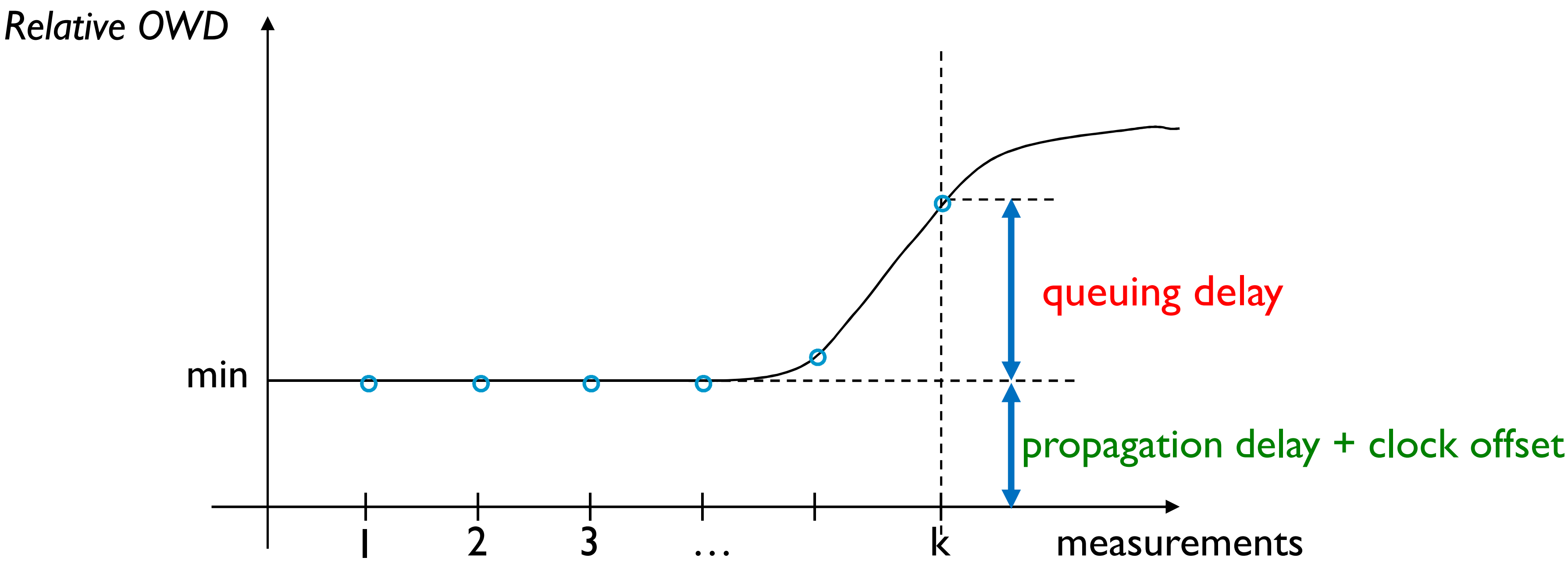
Each packet contains a ***timestamp*** and a ***tree number***

Sending & Forwarding Video





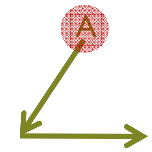
Estimating Queuing Delay Based on Relative *One Way Delay* (OWD) Measurements



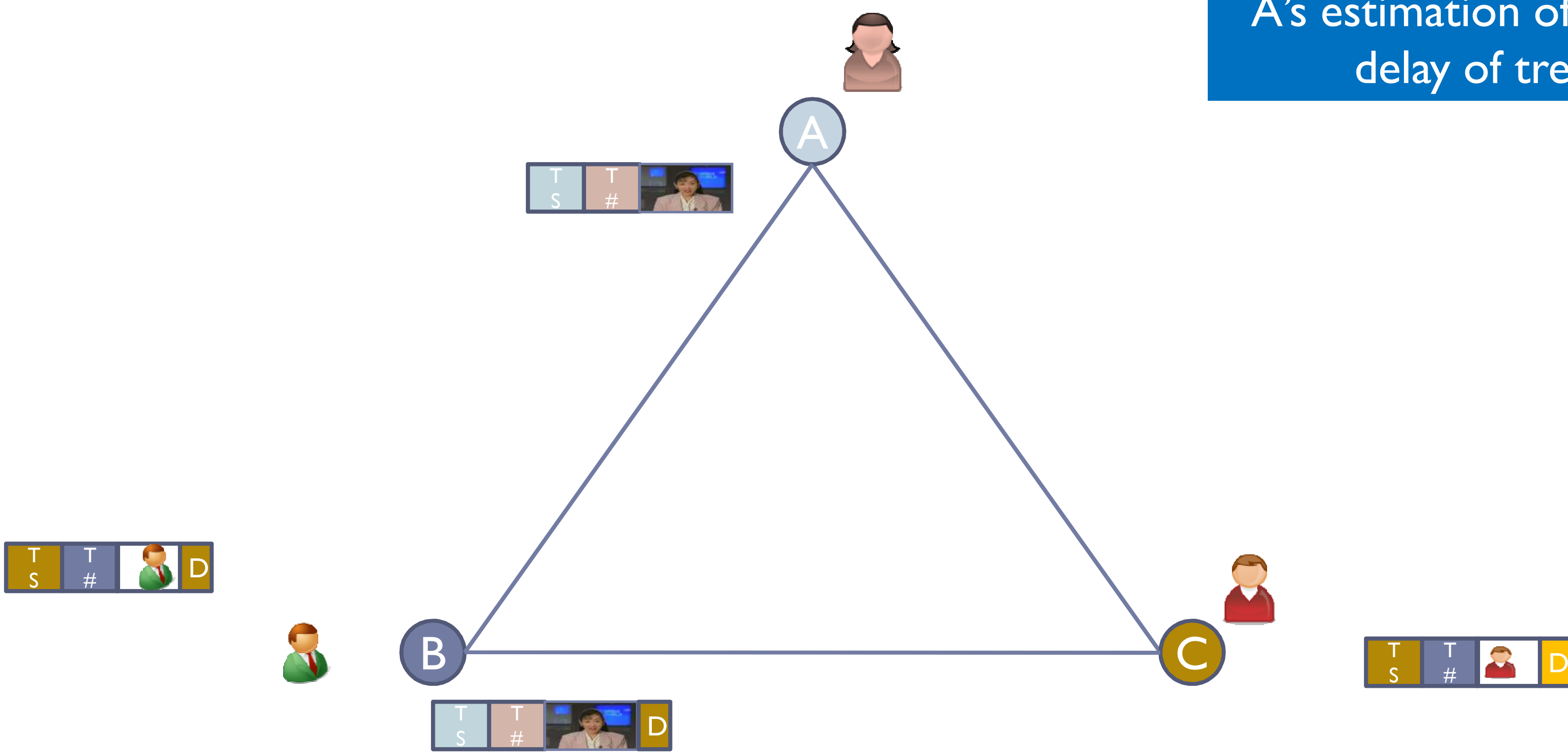
Relative OWD = propagation delay (constant) + clock offset (constant) + queuing delay (variable)

No clock synchronization across peers

Queuing delay information piggybacked to video packets



A’s estimation of queuing delay of tree 2



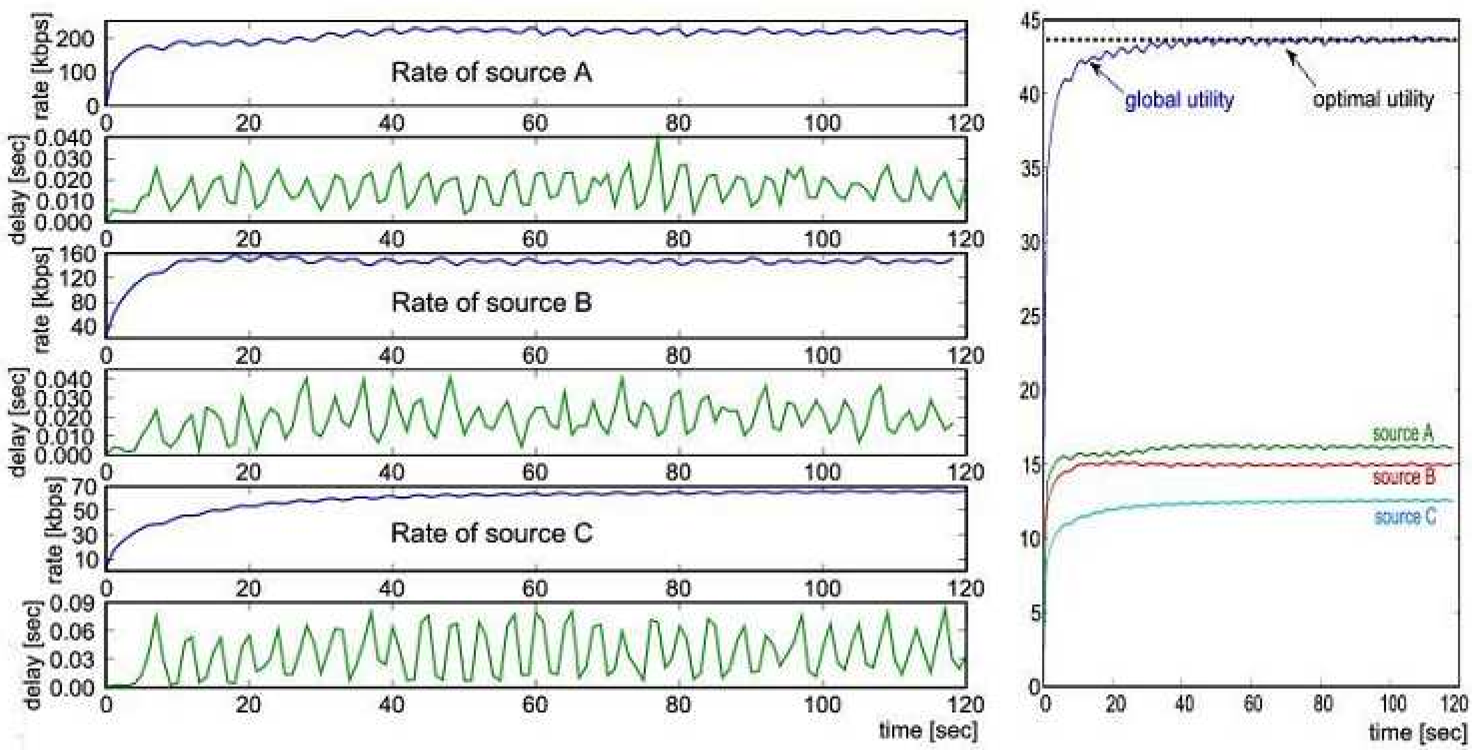
Compute relative OWD between A and B

Compute relative OWD between B and C

An OWD report at most hops one extra peer (helper case)

Internet experiments

- Three peers across US continental: Bay area, Illinois, NYC
  - Uplink capacities: 384, 256, 128 Kbps
  - Estimated one way delay: 40, 20, 33 ms
  - Average packet delivery delay: 95, 105, 128 ms



Concluding Remarks

- Framework and solution for utility maximization in P2P systems
  - Packing linear number of trees per source is optimal in P2P topology
  - Tree-rate based formulation results in linear constraints
- Distributed algorithms for determining source rates and tree splitting
  - Packet marking based primal algorithm
  - Queueing delay based primal-dual algorithm
- Practical implementation of primal-dual algorithm and Internet experiments