Scheduling Multicast Traffic with Deadlines in Wireless Networks

Kyu Seob Kim, **Chih-ping Li**, and Eytan Modiano

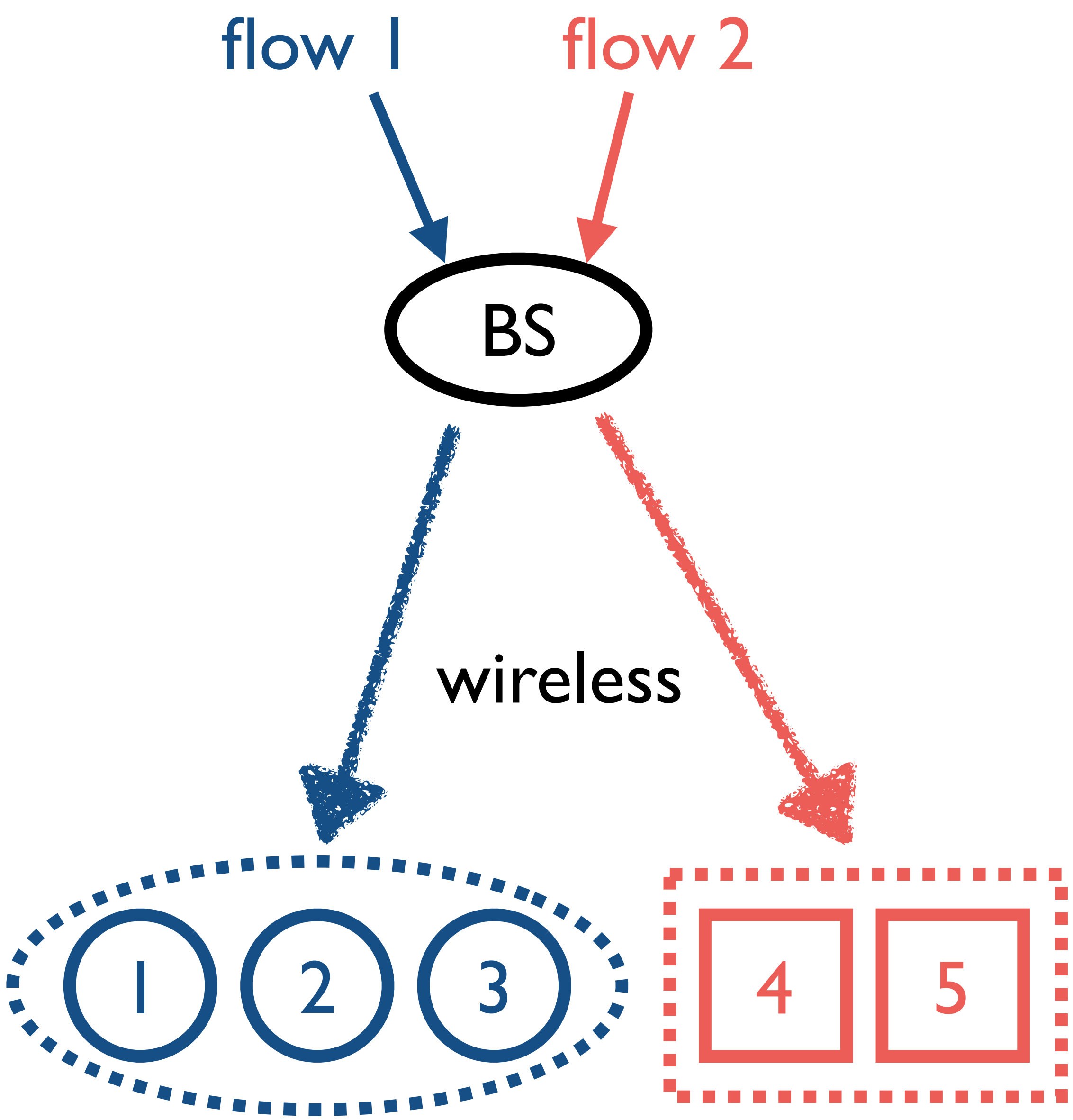
Laboratory for Information and Decision Systems
Massachusetts Institute of Technology

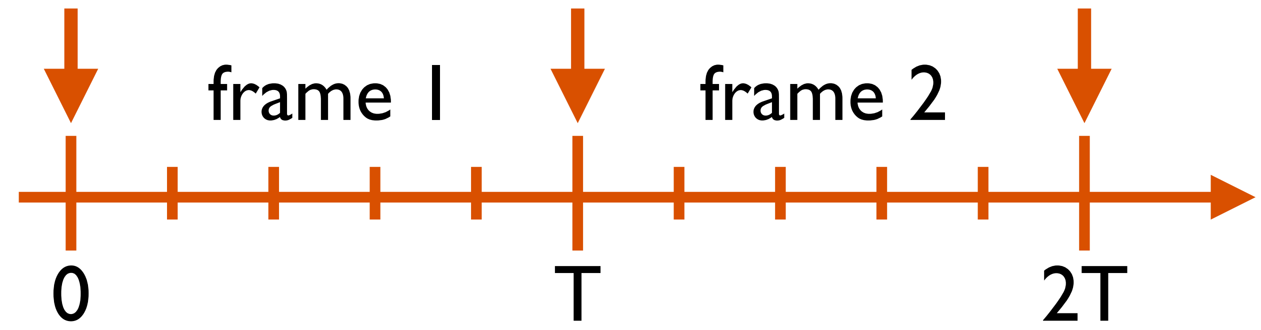
IEEE INFOCOM 2014, Toronto, Canada

Real-time multicast over wireless

- Increasing demand for serving real-time flows in wireless networks
- Challenges:
    - Hard delay constraints
    - Unreliable wireless channels
    - Per-user QoS (throughput) demand
    - Exploiting multicast
    - Exploiting instant user feedback
- We study multicast transmissions with hard deadlines over unreliable wireless channels

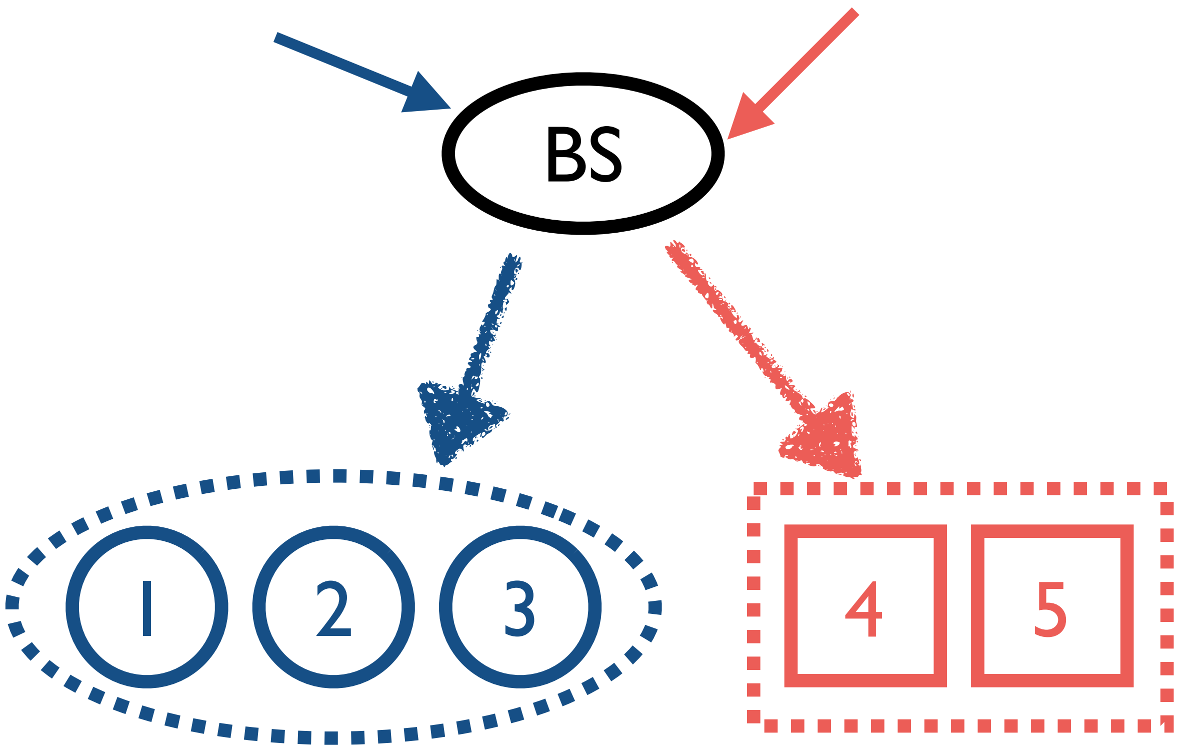
Wireless network model



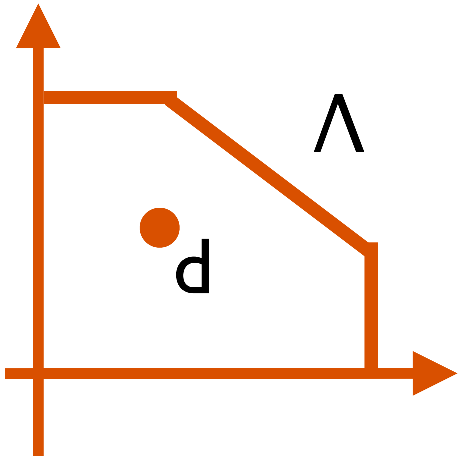


- Each flow generates one packet every frame
- Old packets are discarded at the end of a frame
- BS broadcasts a flow to its subscribers in a slot
- User n receives a packet in a slot successfully with prob. p_n
- Instant ACK/NACK feedback from users to BS

The multicast scheduling problem



- d_n = timely throughput of user n
- Timely throughput region?
- Throughput-optimal policies?

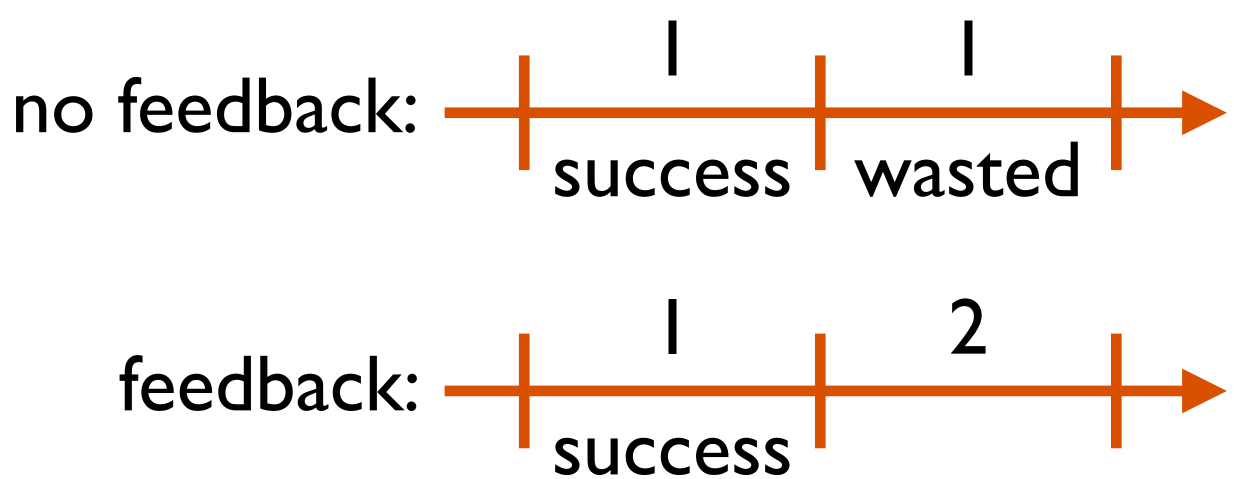


Special examples

- Unicast transmissions [Hou Borkar Kumar 2009]
    - This year's keynote
    - Timely throughput region is a polymatroid [Hou et al 2011]
    - Not applicable to multicast

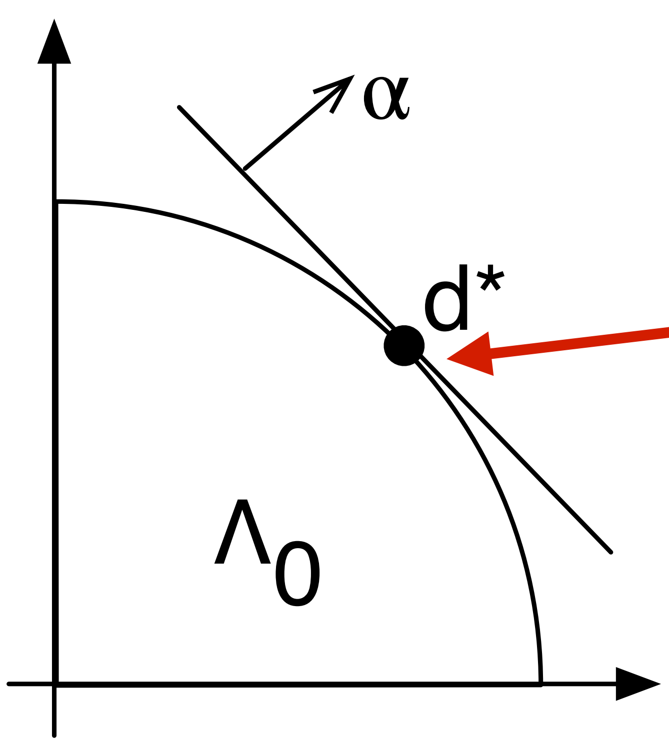
Special examples

- Multicast without user feedback [Hou Kumar 2011]
    - Blind and redundant packet transmissions
    - Throughput region is decided by # of slots allocated to each flow
- Feedback improves throughput



Multicast throughput

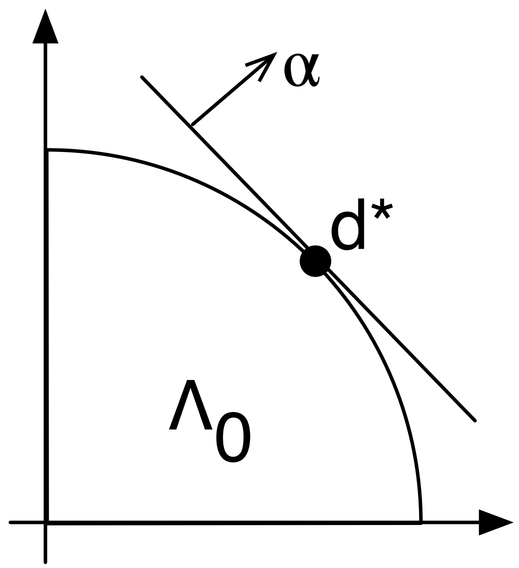
- Throughput region Λ_0 in the first frame
- Computing Λ_0 is difficult
  - finite-horizon DP with exponentially large state space
- We characterize boundary points of Λ_0 by solving:



$$\text{maximize} \sum_{n=1}^{N} \alpha_n \, \mathbb{E}[d_n^\pi(0)], \quad \text{subject to } \pi \in \Pi,$$

(equivalently, finding a supporting hyperplane)

Dynamic programming formulation



$$\text{maximize} \sum_{n=1}^{N} \alpha_n \, \mathbb{E}[d_n^{\pi}(0)], \quad \text{subject to } \pi \in \Pi,$$

State space S = the collection of all subsets of users

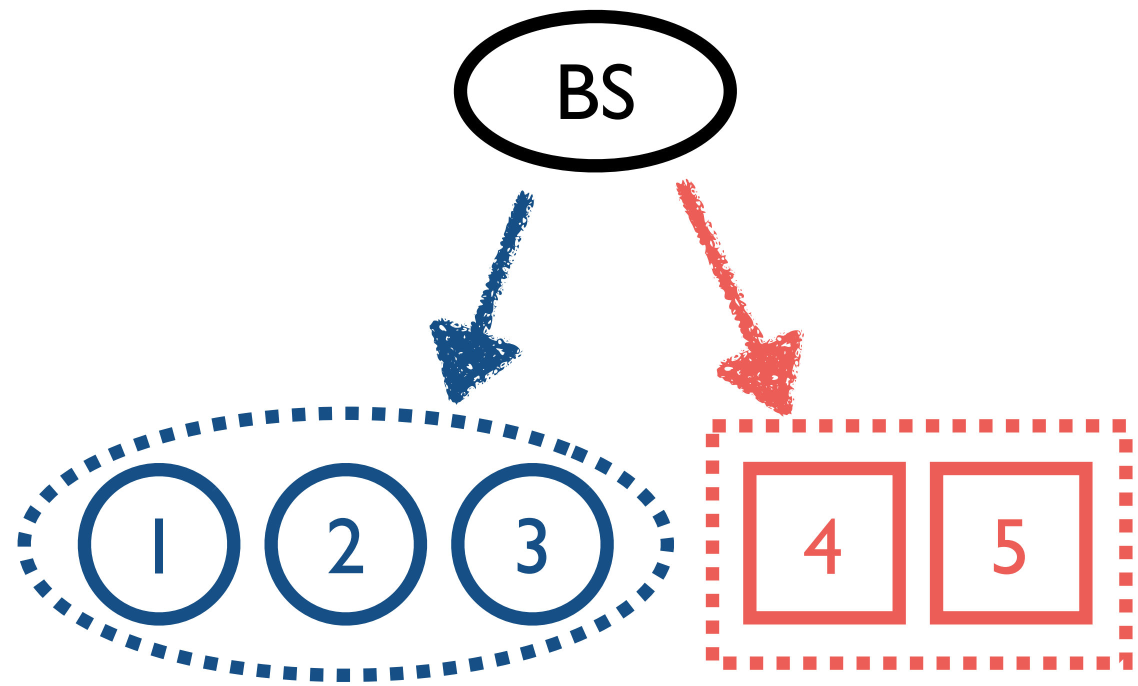
State s_k = subset of unserved users of all flows in slot k

Action $u_k(s_k)$ = multicast flow to serve

Reward $g_k(s_k, u_k)$ = sum of successful transmissions to unserved users, weighted by $\boldsymbol{\alpha}$

Solve: $$J^*(s_0) \triangleq \max_{\pi \in \Pi} \mathbb{E} \left\{ \sum_{k=0}^{T-1} g_k(s_k, u_k) \;\middle|\; s_0 \right\},$$

There is an optimal greedy policy



pi = reward if user i receives a new packet

αi = user-i weight

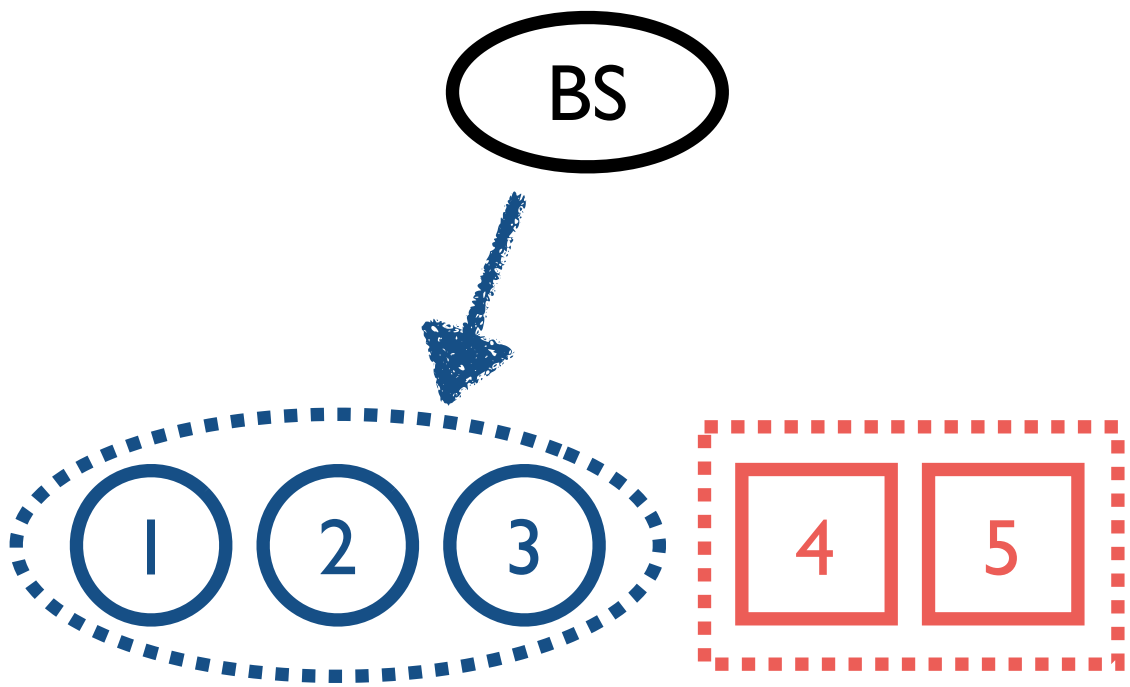
$r_f(t) = \sum_{i \in U_f(t)} \alpha_i \, p_i$: weighted reward from unserved users of flow f

Optimal greedy policy: serve the flow f that maximizes rf(t) in slot t
(it follows backward induction and interchange arguments of DP)

$T = 3$ slots, $t = 1$

$p1 = p2 = p3 = 0.5$

$\alpha 1 = \alpha 2 = \alpha 3 = 2$



$p4 = p5 = 0.4$

$\alpha 4 = \alpha 5 = 3$

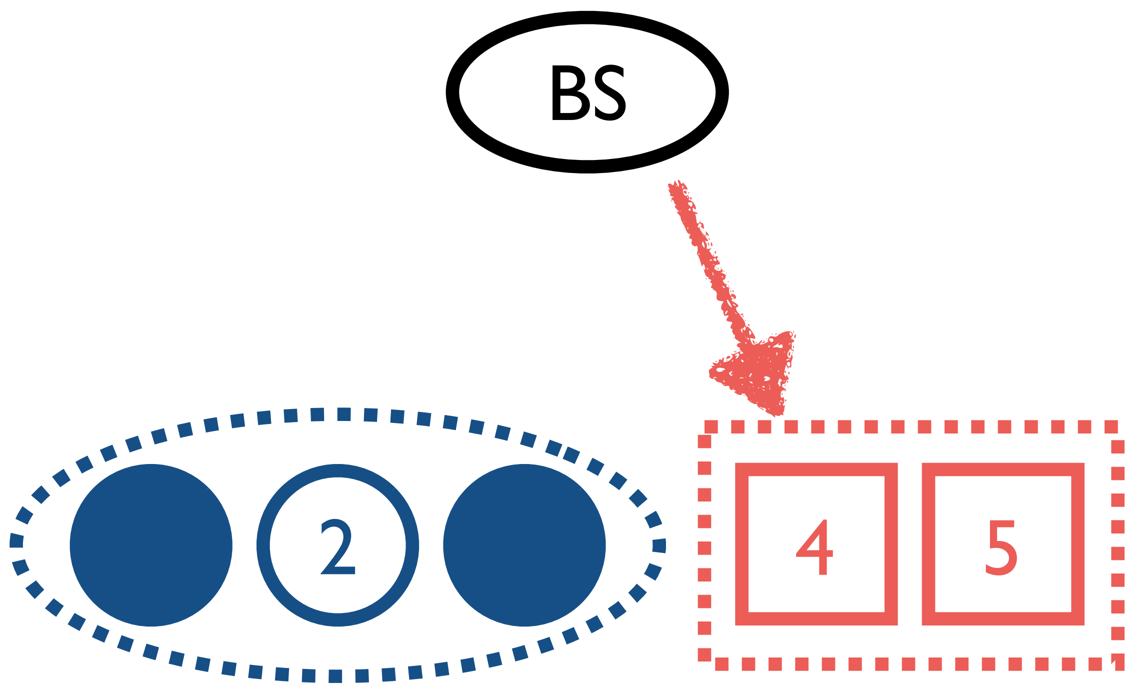
$r1(t) = 0.5 \times 2 \times 3 > r2(t) = 0.4 \times 3 \times 2$

Serve multicast flow BLUE

$T = 3$ slots, $t = 2$

$p1 = p2 = p3 = 0.5$

$\alpha 1 = \alpha 2 = \alpha 3 = 2$



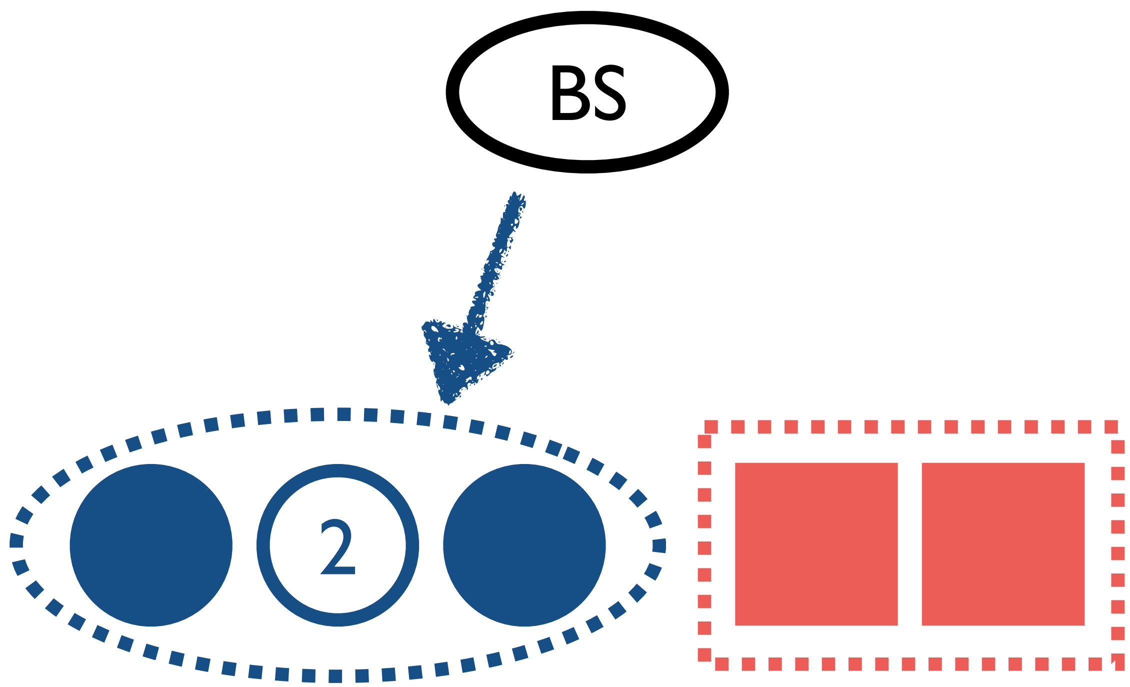
$p4 = p5 = 0.4$

$\alpha 4 = \alpha 5 = 3$

$$r1(t) = 0.5 \times 2 \times 1 < r2(t) = 0.4 \times 3 \times 2$$

Serve multicast flow RED

T = 3 slots, t = 3



p1= p2 = p3 =0.5

α1 = α2 = α3 = 2

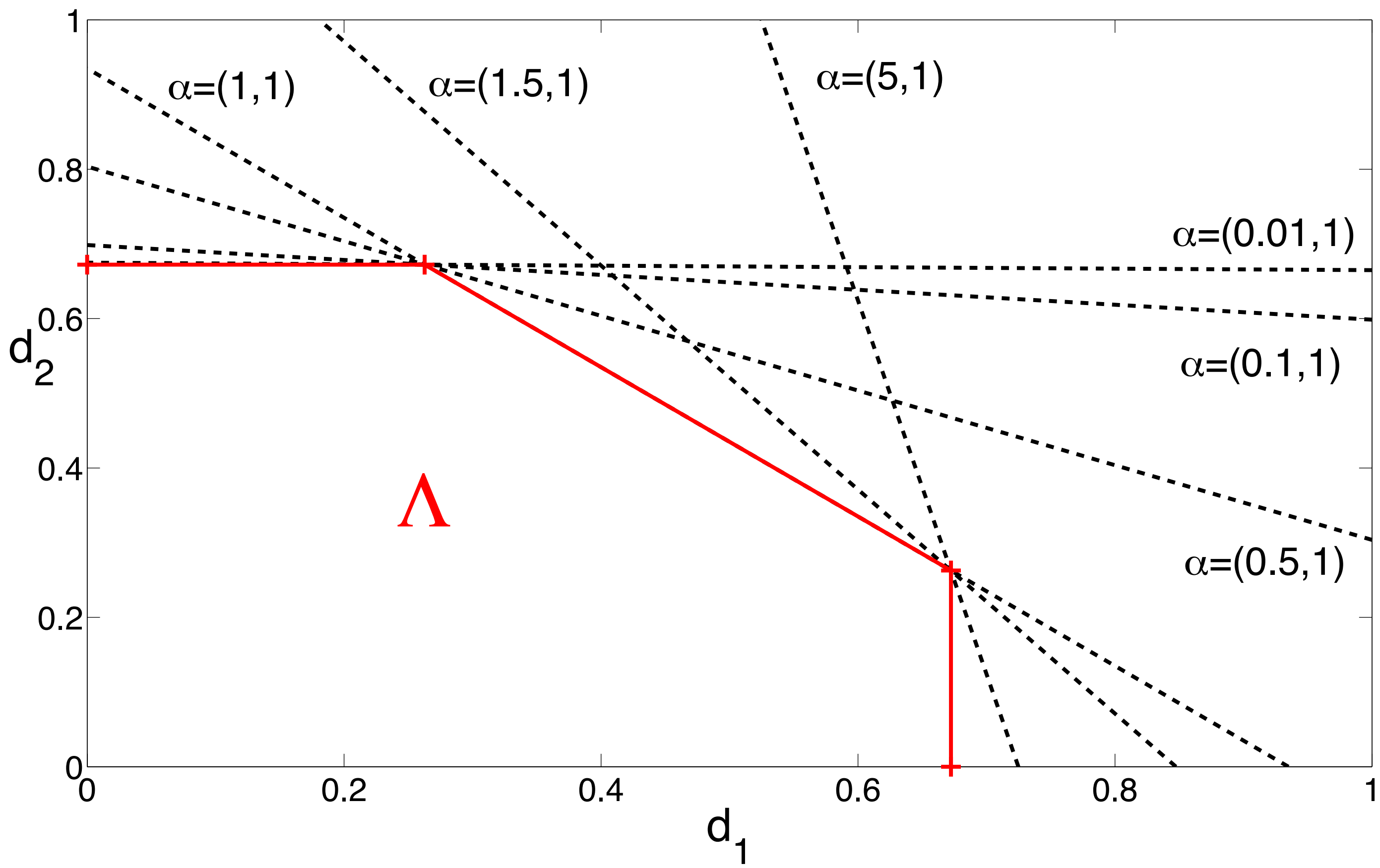
p4 = p5 = 0.4

α4 = α5 = 3

r1(t) = 0.5 x 2 > r2(t) = 0

serve multicast flow BLUE

Special example: unicast transmissions



Two users
Frame size T=5 slots
Channel probabilities p1=p2=0.2

Throughput region is given by supporting hyperplanes

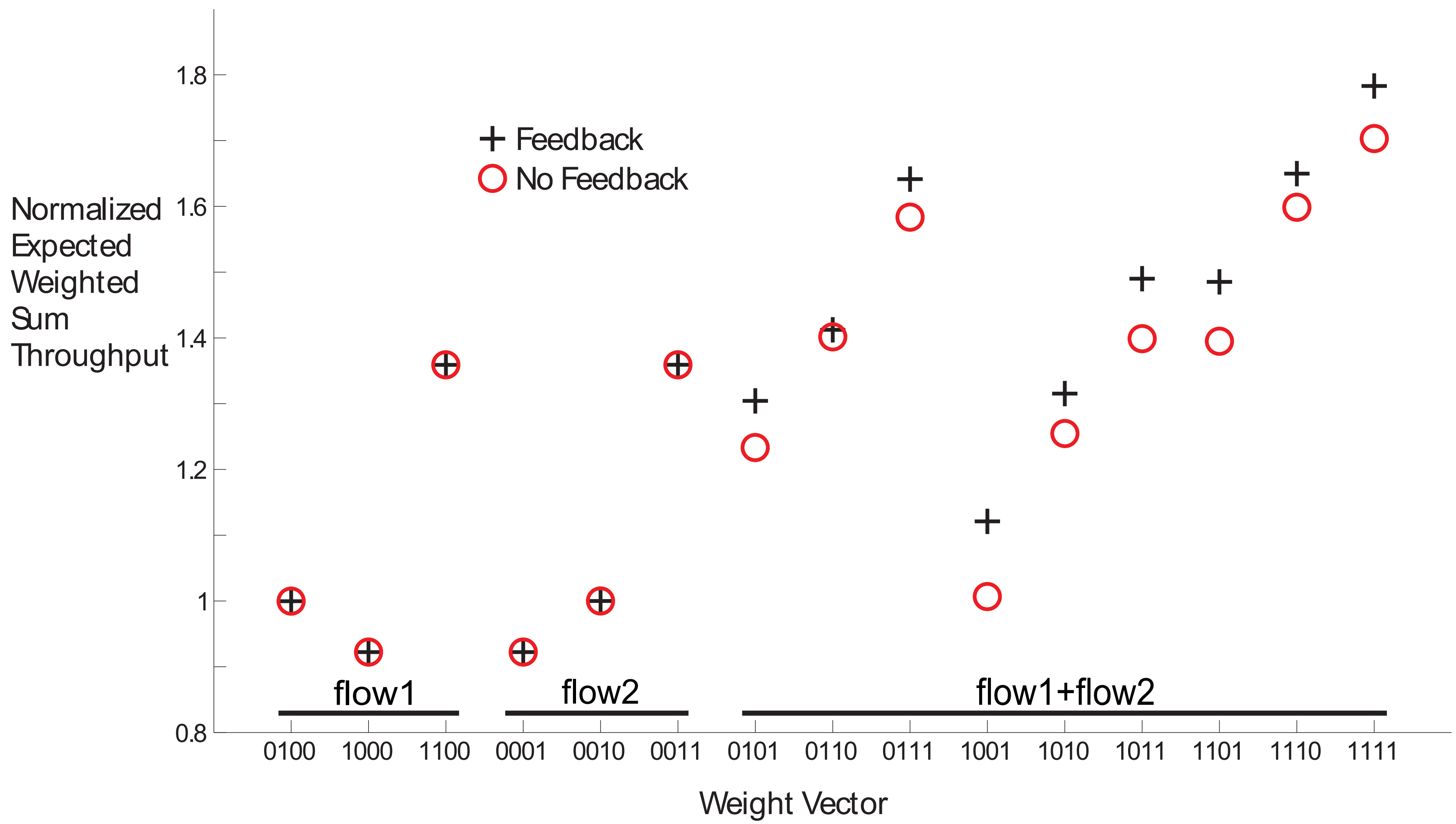
[Hou Borkar Kumar 2009]

Throughput-optimal policy

- Achieving feasible throughput q_n for each user n
- Throughput-optimal policy:
  - Delivery debt up to frame k: $$D_n(k) = \sum_{j=0}^{k-1} (q_n - d_n(j))$$
    $$D_n^+(k) = \max\{D_n(k), 0\}$$
  - Transmit the flow f that maximizes rf(t) from unserved users in slot t of the kth frame:

$$r_f(t) = \sum_{n \in U_f(t)} \underbrace{D_n^+(k)}_{\text{reducing long-term debt}} \cdot \underbrace{p_n}_{\text{maximizing immediate reward}}$$

Feedback improves throughput



Users 1 & 2 subscribe to flow 1 with (p1, p2) = (0.4, 0.8)

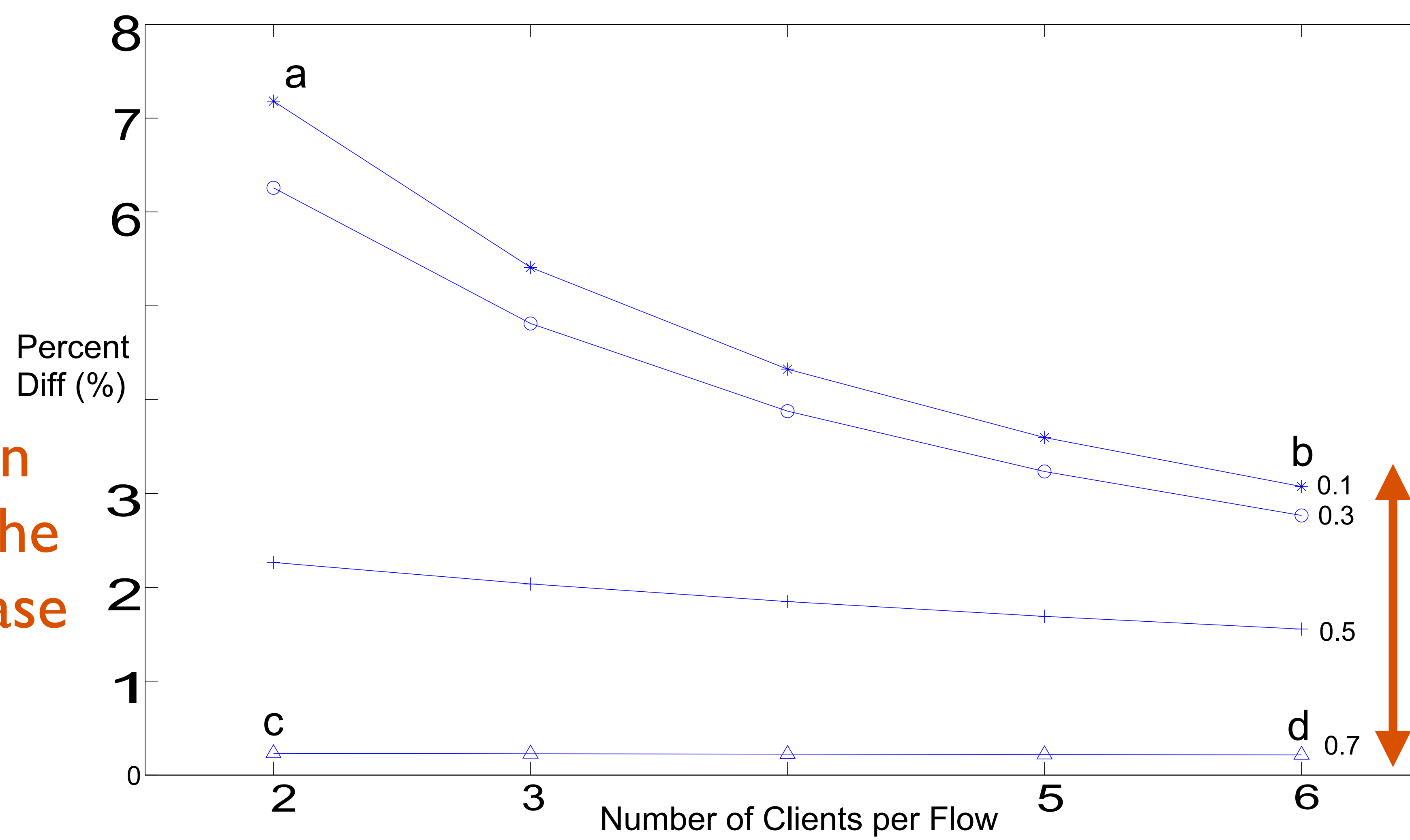
Users 3 & 4 subscribe to flow 2 with (p3, p4) = (0.9, 0.4)

Frame size T=5

Popularity of multicast flows

2 multicast flows, frame size T=10
All users have the same channel probability p

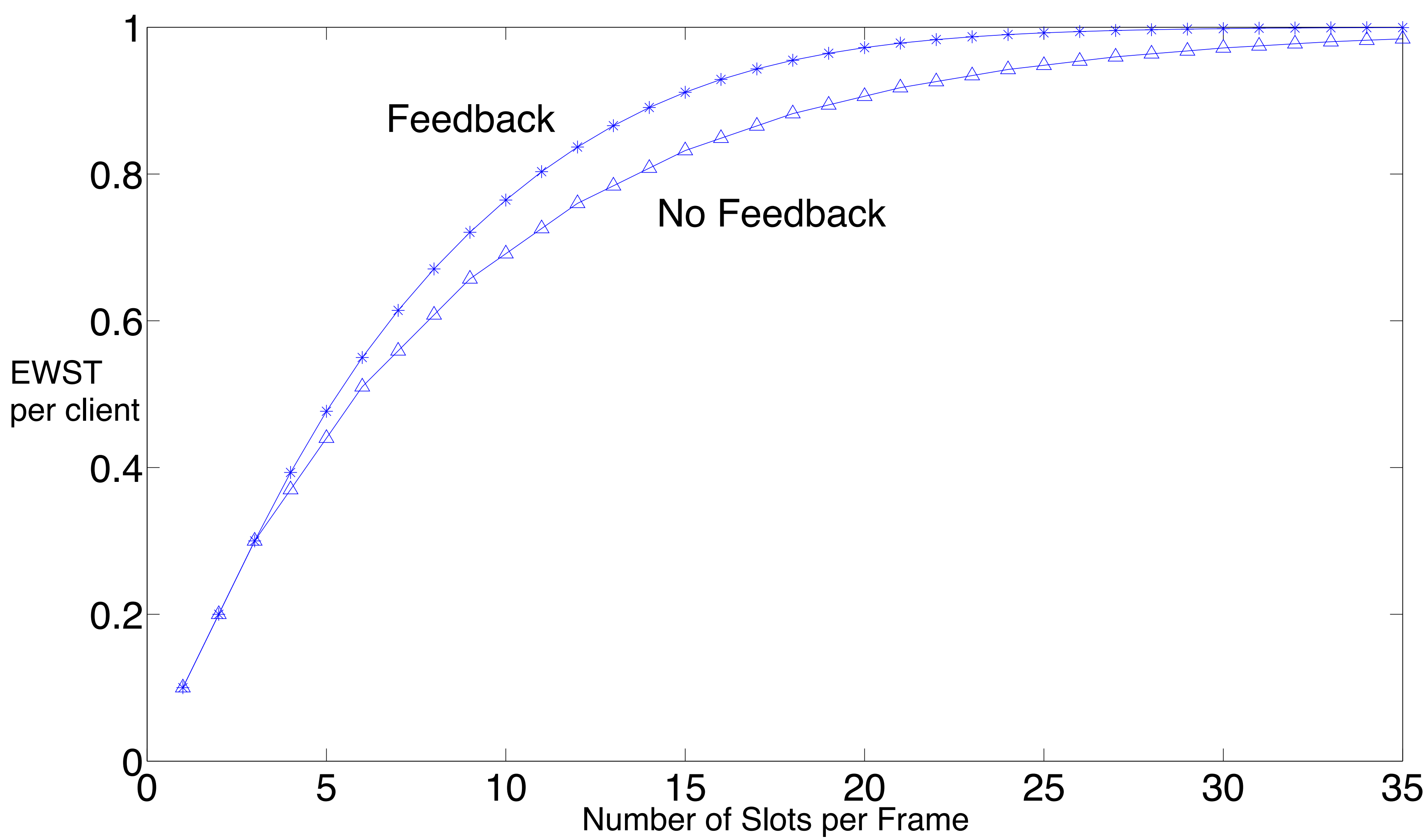
throughput gain compared to the no-feedback case



different p

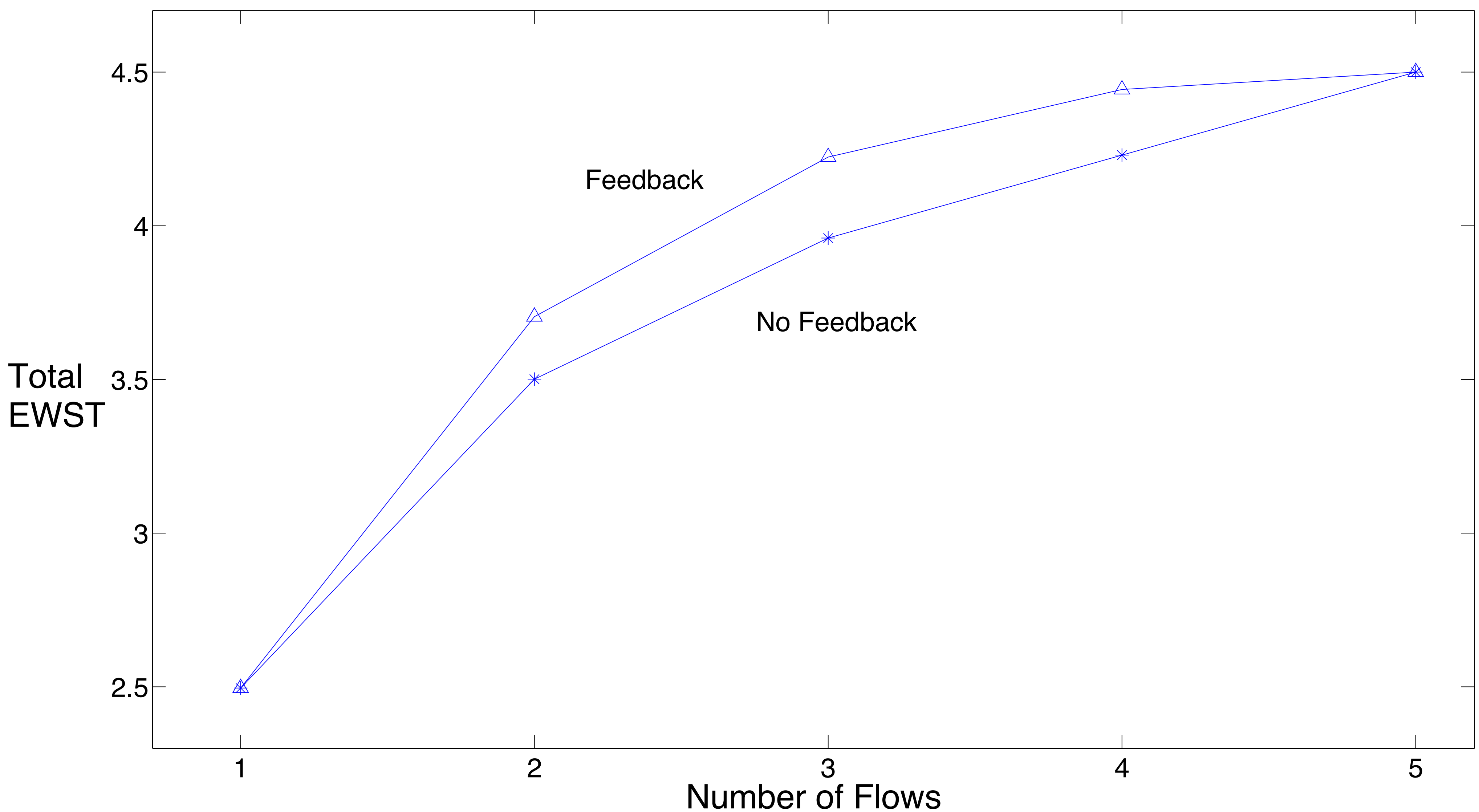
Different frame sizes

3 multicast flows, each has 2 subscribers
All users have the same channel probability p=0.3



Number of multicast flows

Each multicast flow has 3 subscribers, frame size T=5
All users have the same channel probability p=0.3



Summary

- Multicast is efficient for real-time transmissions over wireless
    - Issues include unreliable channels, hard deadline constraints, and per-user QoS demand, and multicast protocol design
- When user feedback is available, the multicast throughput region is characterized by greedy policies
    - Using backward induction and inter-change arguments of DP
- An adaptive throughput-optimal policy
- Simulations identify conditions under which user feedback provides the largest and the lowest throughput gain over the no-feedback case