FAST Copper For Broadband Access: An Overview

Mung Chiang
Electrical Engineering Department, Princeton
With J. Huang, D. Xu, Y. Yi, C. W. Tan, R. Cendrillon

www.princeton.edu/fastcopper

Allerton, SPIE, INFORMS
September – November, 2006

What's FAST Copper?

>10X improvement in copper-last-mile broadband access through fiber/DSL deployment, engineering innovations, and fundamental research

R3Q: Rate (at application level), reach, reliability, quality

- NSF ITR sponsorship
- Princeton, Stanford, Fraser Research Lab
- PI: M. Chiang, Co-PIs: J. Cioffi and A. Fraser
- Main industry collaborator: AT&T

Timeline:

- 2002-2004: initial work with SBC
- 2004-2008: formal duration of the project
- 2008-: continued research and industry adoption

Outline

- Is it possible to get truly broadband with phone line?
- Architectural issues
- Frequency
- Amplitude
- Space
- Time

FAST and FAST: FAST Copper is different from TCP FAST

Research talk: Not focusing on stories about industry deployment

Midway report: FAST Copper is just starting to gain full momentum

Partial report: Only Princeton's part summarized here

Introduction

Why Fiber/Copper?

Alternatives of broadband access:

- Wireless: reliability, coverage, and backhaul issues
- Cable modem: not ubiquitous, bandwidth sharing issues
- Fiber to the closet: per-customer labor cost prohibitive (especially for "brown-field" suburban in US)
- Existing DSL: 160 million users, but not fast enough

- Fiber/Copper: Best of ubiquity, broadband, reliability, and migration

Broadband over fiber and phone wires

Example: AT&T's Lightspeed Project

Where Are Bottlenecks and Where To Improve

- Attenuation: Solution from Space
- Crosstalk: Solutions from Frequency, Amplitude, Time

Realistic estimates on improvements coming from research:

- Frequency: 2X (even more through signal processing)
- Amplitude: $>$2X
- Space: enabler of rate, reach, reliability
- Time: 2X

Not even bringing in wider bandwidth, multiple twisted-pairs, and systems debugging yet

Key Ideas

- It’s not a dedicated line, it’s a (multi-carrier) interference channel

Turn competition to cooperation in frequency and time

From “low frequency” mentality to “high frequency” mentality

- It’s not a voice line, it’s a bursty data and video line

Squeeze in more than you have bandwidth for

From “deterministic” mentality to “statistical” mentality

How to make the engineering work?

A lot of research (and deployment) challenges

Challenges and Connections

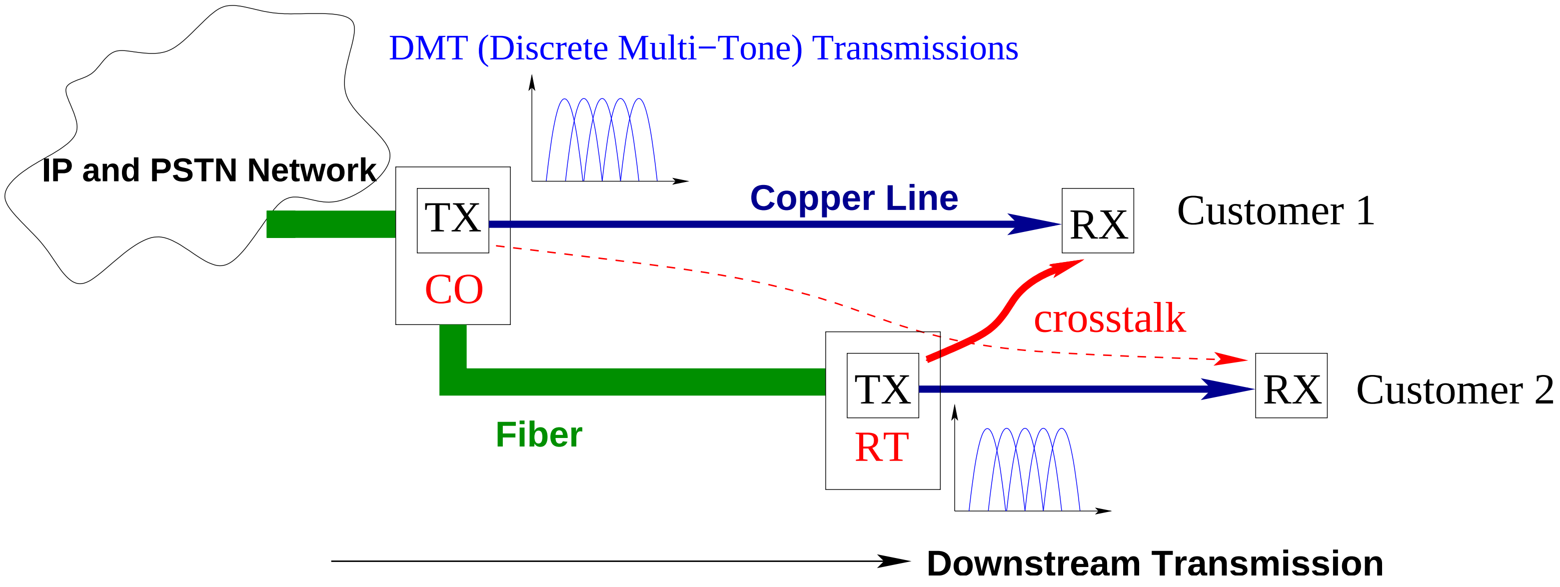
Two types of challenges

Many challenging problems in terms of resource allocation:

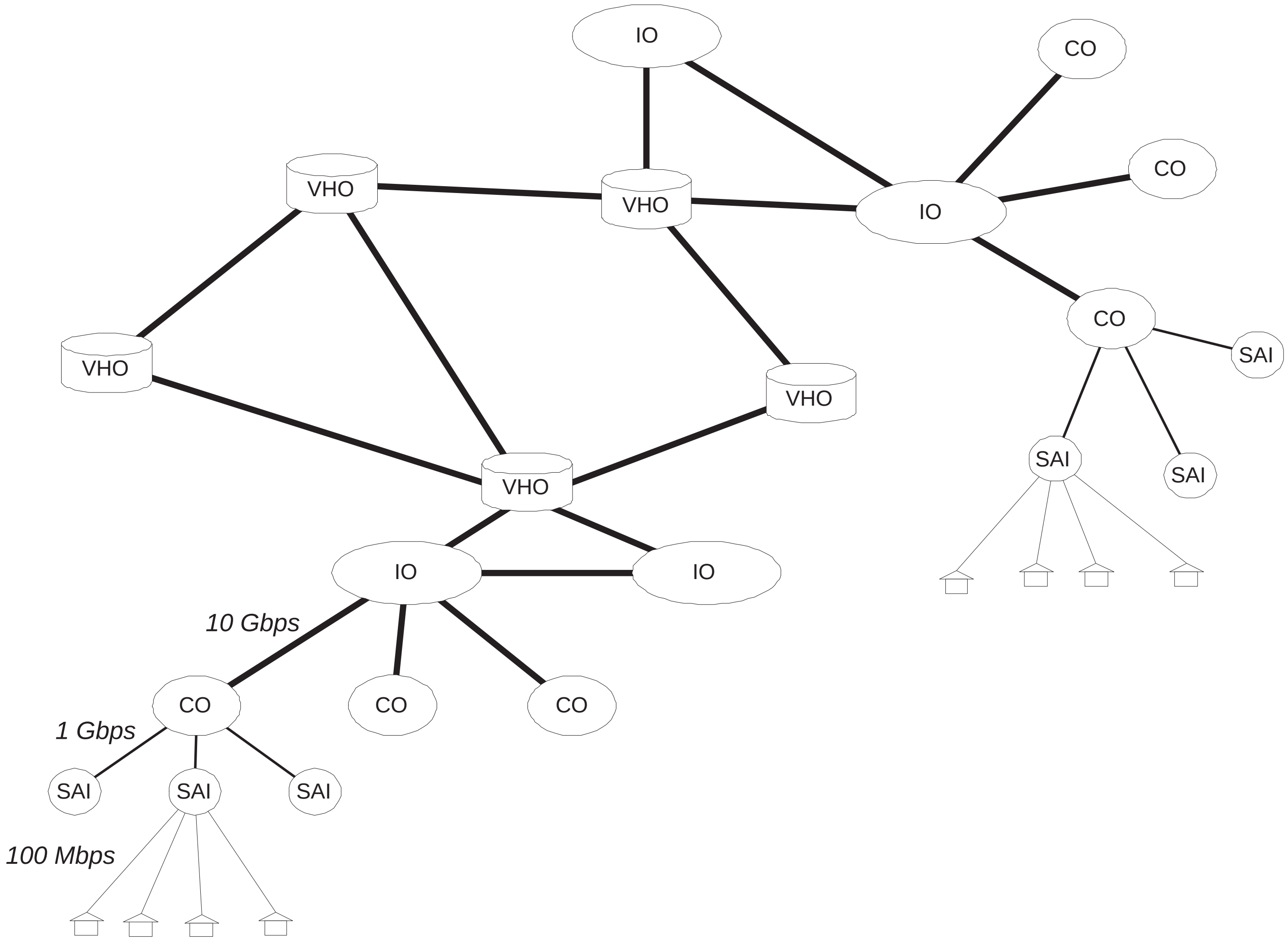
- Information theory: multi-carrier interference channel
- Signal processing: multi-user transmissions
- Stochastic theory: statistical multiplexing
- Graph theory: survivable tree design
- Optimization theory: nonconvex and globally coupled optimization
- Networking: resource allocation and “Layering As Optimization Decomposition”

But the biggest challenge is architecture design for broadband access

Typical Deployment: Access Part



Typical Deployment: End-to-end



Architectural First

- Architecture: functionality allocation

More influential, harder to change, less understood than resource allocation

Metrics: Performance, X-ities, Cost and complexity

- Modularization: vertical decomposition by a protocol stack
- Distribution of control: horizontal decomposition into network elements

- Coupling between horizontal and vertical decompositions

Example: who takes care of traffic shaping?

Example: Where to do error control: FEC, ARQ, R-UDP, TCP, or application layer?

Horizontal Decomposition

- Video server placement:

Tradeoff between response time and scalability

- Distribution server and cache placement:

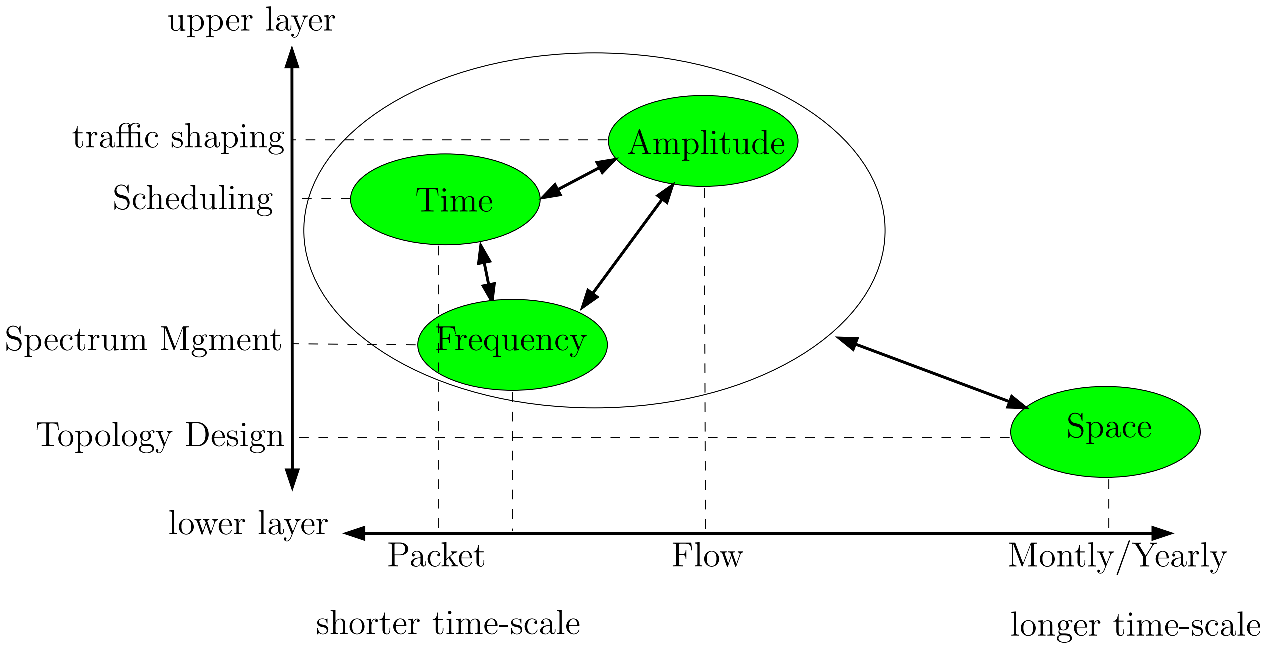
Where to take care of channel changes?

Where are the boundaries of multicast group?

- Even bigger issue: How big should the access network be?

Tradeoff among reliability of access tree, feasibility of big switches, complexity of backbone network, ease of management

Vertical Decomposition and Time-Scales of F A S T



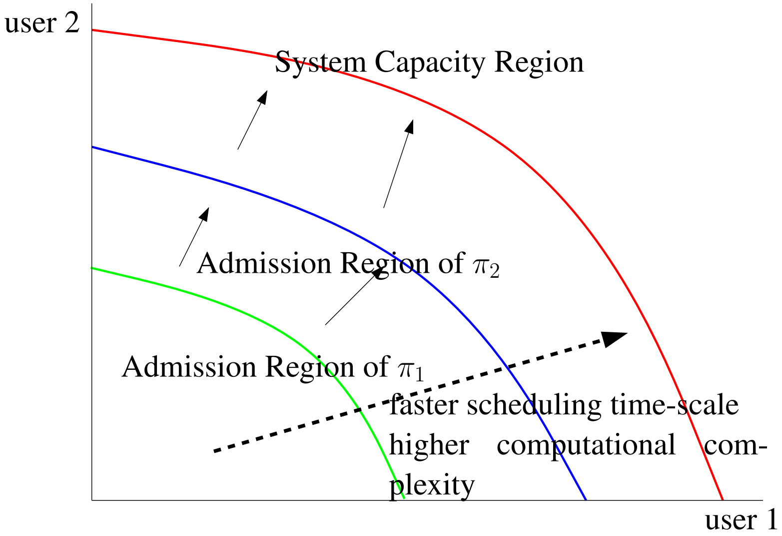
- Time-scale: Time $>$ Frequency $>>$ Amplitude $>>$ Space
- Low-complexity Spectrum Management Algorithm: Time $\approx$ Frequency
- Time-scale separation lowers price of modularity

Vertical Decomposition and Time-Scales of F A S T

Extreme cases: spatial division multiplexing (S), time division multiplexing (T), frequency division multiplexing (F), turn away users (A) can all tackle crosstalk

- Possible rate regions attainable (Frequency): determined by deployment topology (Space)
- Feasibility and stability of scheduling (Time): determined by placement of traffic shapers and schedulers (Space)
- Two obviously coupled degrees of freedom: Time and Frequency
- Furthermore, capability of Time: determined by time-scale of Frequency
- Amplitude control depends on rate region attainable (Frequency)
- Interesting interaction between Time and Amplitude: next slide

Modularity-Performance Tradeoff



- $\mathcal{A}(\pi_1) \subset \mathcal{A}(\pi_2)$, where $\mathcal{A}(\pi)$: admission region of scheduling π
- Scheduling Algorithm: π_1 and π_2
  - π_1: at flow-level time-scale
  - π_2: at packet-level time-scale exploiting opportunism
- Conservative admission control $\mathcal{A}(\pi_1)$ removes the need for π_2 scheduling

Mid-point in the Talk

- Move from the quantification of architectural tradeoffs to
- A very brief summary of current progress on F, A, S, T

Frequency

Dynamic Spectrum Management

Question: How to allocate power (bit loading) across different tones and competing users to turn competition to cooperation?

Problem formulation:

$$\begin{array}{ll} \underset{\{\boldsymbol{p}_n \geq \mathbf{0}\}_n}{\text{maximize}} & \sum_n w_n R_n \\ \text{subject to} & \sum_k p_n^k \leq P_n, \forall n \end{array}$$

- User n's achievable rate $R_n = \sum_k \log\left(1 + \frac{p_n^k}{\sum_{m \neq n} \alpha_{n,m}^k p_m^k + \sigma_n^k}\right)$
- Total power constraint: $\mathcal{P}_n = \left\{p_n^k \geq 0, \forall k, \sum_k p_n^k \leq P_n^{\max}\right\}$
- Characterize Pareto boundary of rate region [Centrillon et. al. 04]

Challenging optimization problem: Nonconvex and coupled (across users and across tones)

History of DSM algorithms

- IW: Iterative Water-filling [Yu Ginis Cioffi 02]
- OSB: Optimal Spectrum Balancing [Cendrillon et. al. 04]
- ISB: Iterative Spectrum Balancing [Liu Yu 05] [Cendrillon Moonen 05]
- ASB: Autonomous Spectrum Balancing [Huang Cendrillon Chiang Moonen 06]

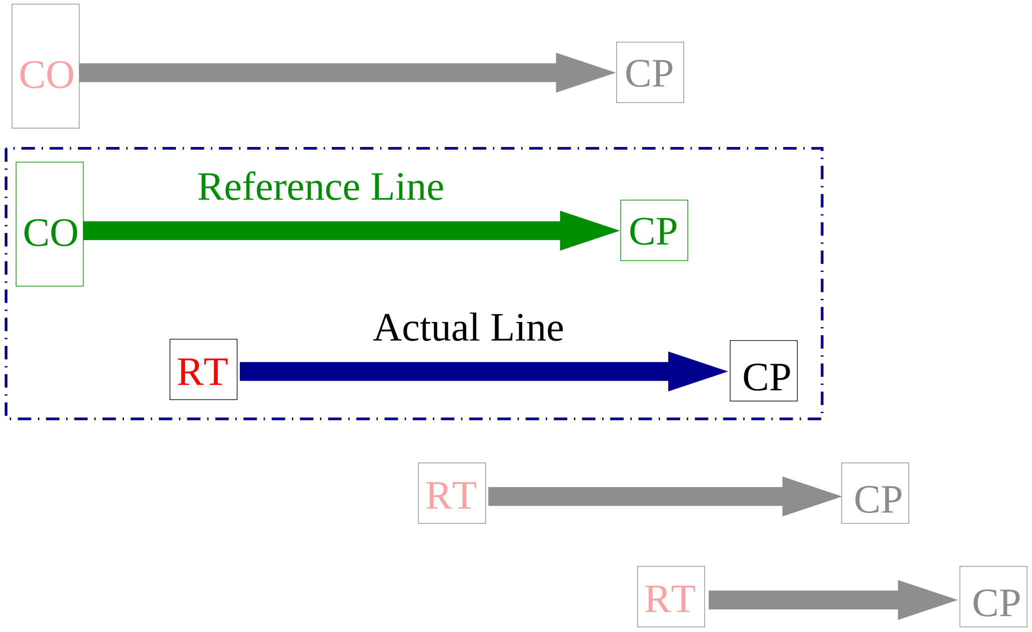
Algorithm	Operation	Complexity	Performance
IW	Autonomous	$O\left(KN\right)$	Suboptimal
OSB	Centralized	$O\left(Ke^{N}\right)$	Optimal
ISB	Centralized	$O\left(KN^{2}\right)$	Near Optimal
ASB	Autonomous	$O\left(KN\right)$	Near Optimal

K: number of carriers N: number of users

Reference Line Concept

Dynamic pricing for dynamic coupling: decouple tones

Static pricing for static coupling: decouple users



Key Idea of ASB

- User n solves the following problem:

$$\begin{aligned} \underset{\boldsymbol{p}_n \geq \mathbf{0}}{\text{maximize}} \quad & w_n R_n + R_n^{\text{ref}} \\ \text{subject to} \quad & \sum_k p_n^k \leq P_n \end{aligned}$$

where the reference line rate is:

$$R_n^{\text{ref}} = \sum_k \log\left(1 + \frac{p^{k,\text{ref}}}{\alpha_n^{k,\text{ref}} p_n^k + \sigma^{k,\text{ref}}}\right)$$

- Parameters in red are constants known a priori through channel measurement
- Autonomous: Only local information is needed
- Low complexity and achieve near optimal performance

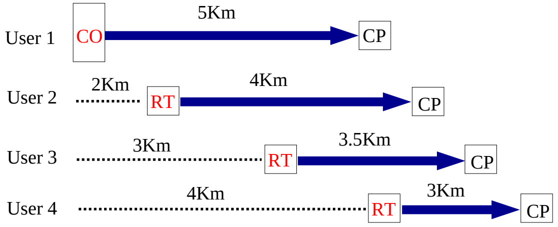
ASB Algorithm: Basic Sketch

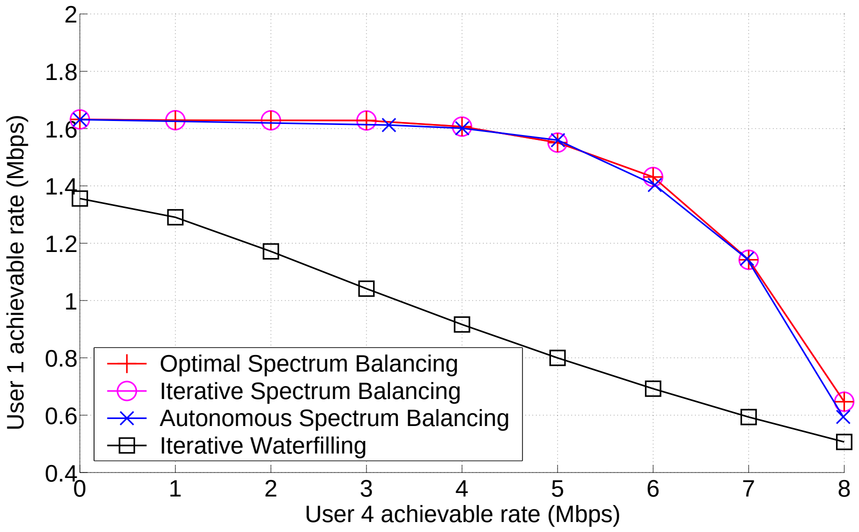
```
repeat
    for each user $n = 1, ..., N$
        repeat
            for each carrier $k = 1, ..., K$, find
                Find $p_n^k$ by solving one subproblem for tone $k$
            $\lambda_n = \left[\lambda_n + \varepsilon_\lambda \left(\sum_k p_n^k - P_n^{\max}\right)\right]^+$
            $w_n = \left[w_n - \varepsilon_w \left(\sum_k R_n^k - R_n^{\text{target}}\right)\right]^+$
        until convergence
    end
until convergence
```

Typical Result from Realistic Simulator

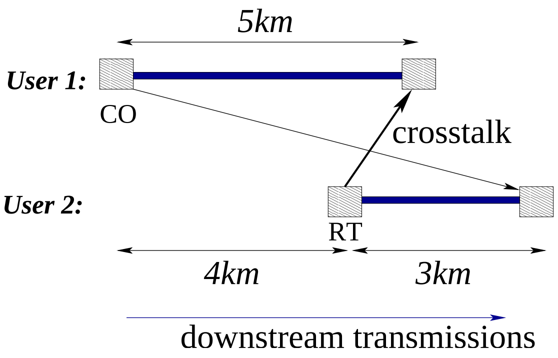
Almost identical to optimal benchmark by centralized computation

More than double the rate for typical deployment scenario





Typical Spectrum



5km

User 1:

CO

crosstalk

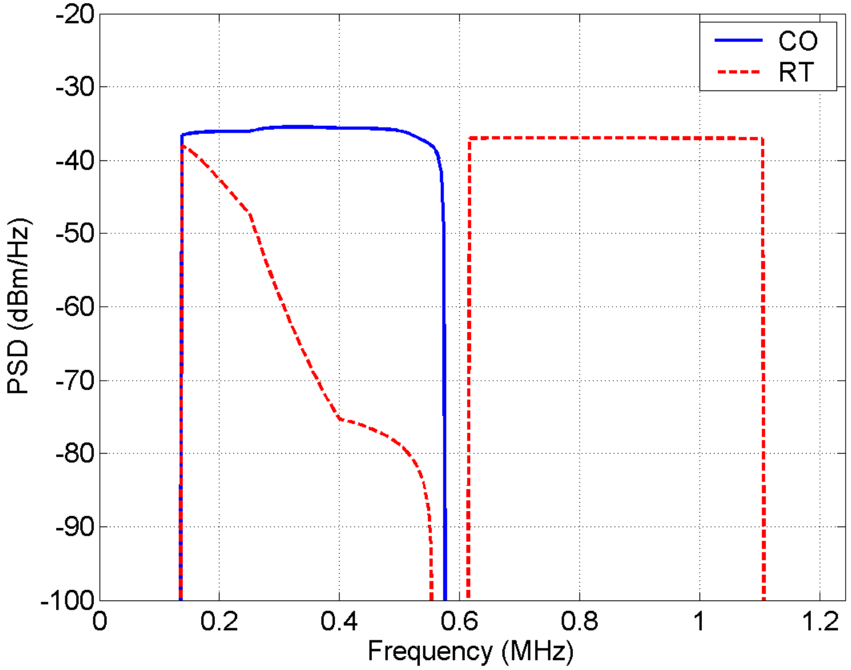
User 2:

RT

4km

3km

downstream transmissions



Convergence Guarantee

Theorem: ASB algorithm (under high SNR approximation, which leads to frequency-dependent waterfilling) converges to the unique fixed point under both sequential and parallel updates, if the crosstalk channels satisfy (physical meaning also obtained):

$$\max_{n \neq m, k} \alpha_{n,m}^{k} < \frac{1}{N-1}$$

- Recover the convergence of iterative water-filling as a special case
- Convergence independent of reference line parameters
- Performance robust to reference line parameters

Extensions:

ASB for asynchronous transmissions with inter-carrier-interference

Amplitude

Multiplexing and Shaping

Question: How aggressive can we exploit burstiness of triple play (voice, data, video) traffic?

- Objective: squeeze maximum number of flows into the network, subject to the statistical QoS requirements
- Previous work in wireline network focus on fixed link rates
- DSL network has the capability to “shuffle” the underlying link rates
- How does this impact the statistical multiplexing decisions? How to do admission control?

Example of Problem Formulation

- Transform stochastic traffic into Effective Bandwidth (EB)
- The value of EB depends on the traffic characteristics, buffer allocation, and QoS requirement

$$\begin{array}{rll}
\text{maximize} & \displaystyle\sum_i w_i a_i n_i & \text{(total weighted throughput)} \\
\text{subject to} & n_i \nu_i \left(\epsilon_i, B_i\right) \leq c_i, \forall i & \text{(EB less than allocated rate)} \\
 & \displaystyle\sum_i B_i = B, & \text{(total buffer constraint)} \\
 & \boldsymbol{c} \in \mathcal{C} & \text{(capacity region constraint)} \\
\text{variables} & \boldsymbol{n}, \boldsymbol{c}, \boldsymbol{B} \geq \mathbf{0} & \text{(\# of flows, rate, buffer)}
\end{array}$$

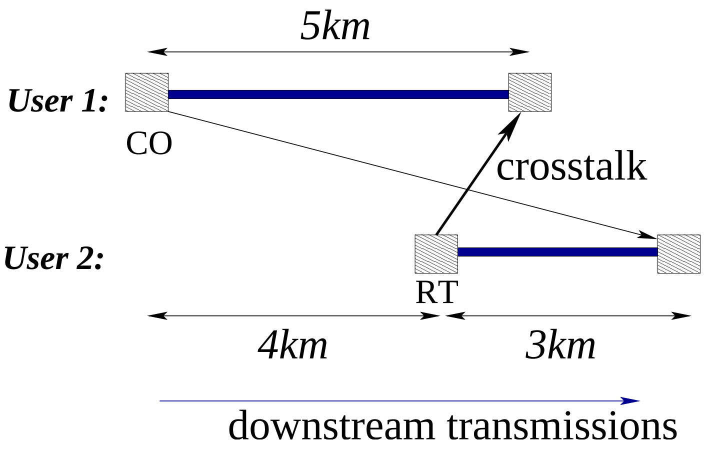
Example of Algorithm

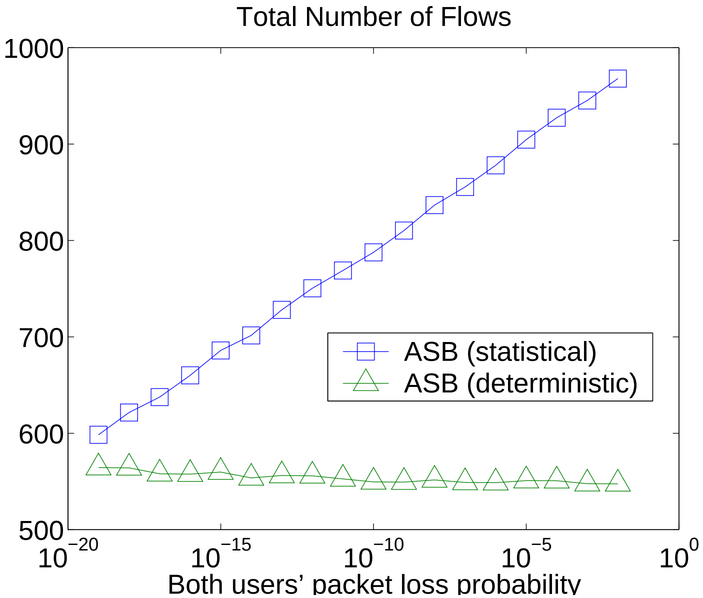
Two-stage Alternate Maximization (AM) Algorithm:

- Rate Allocation stage (for fixed buffer $\boldsymbol{B}$): reduce to weighted rate maximization, can be solved by ASB (autonomous and low complexity)
- Buffer Allocation stage (for fixed rate $\boldsymbol{c}$): reduce to quasi-concave maximization, can be solved by bi-section search (in general needs centralized coordination)
- Alternate through two stages until no further improvement can be obtained

- Theorem: AM algorithm converges

Numerical Example





Space

Overview

- Fat-tree access network topology:

⇒ Make it robust and survivable

⇒ Make it economically viable

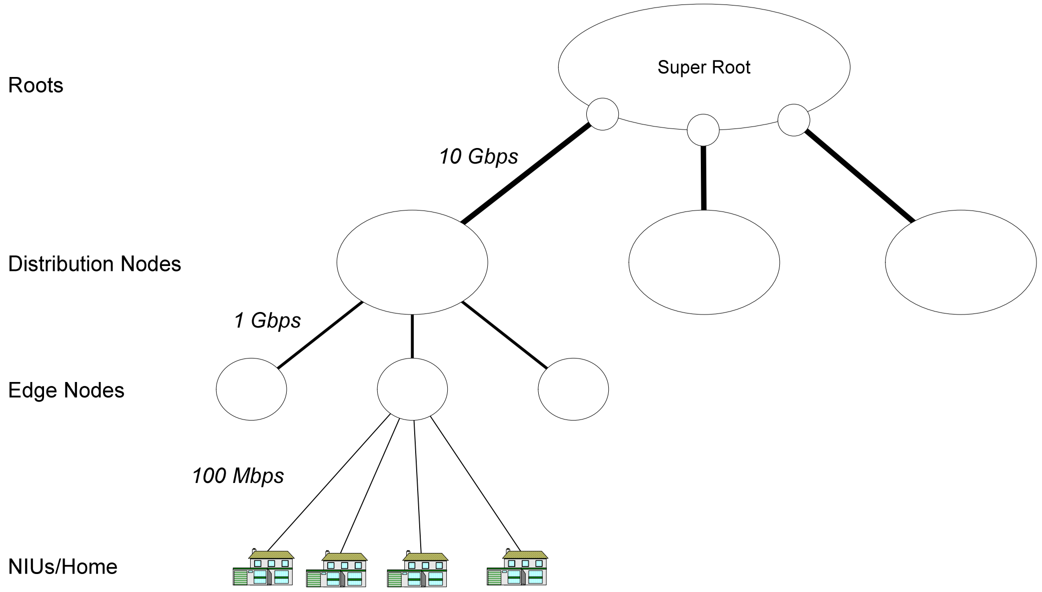
Question: How to add a few links to the tree to make it survivable and economically viable?

Three major components of design:

- Graph theory problem: determine survivable topology (this talk)
- Optimization problem: allocate bandwidth (another talk)
- CS systems problem: design real-time signalling protocol (Fraser Lab)

Fat Tree Topology

Survivable access network design different from backbone network



Logical Fat-Tree Architecture for Access Network

Variations Along Four Dimensions

- Fat-tree exists or not

- Single level or multi-level tree

- Optimization (objective-constraints) model:

⇒ Minimize total cost with connectivity requirement
(e.g. r_i edge-disjoint paths from remote terminal i to root)

⇒ Maximize survivability-based revenue (eg, proportional to number of backup paths) with limited budget (r_i is variable)

- Link cost model:

⇒ Concave edge cost model: buy-at-bulk

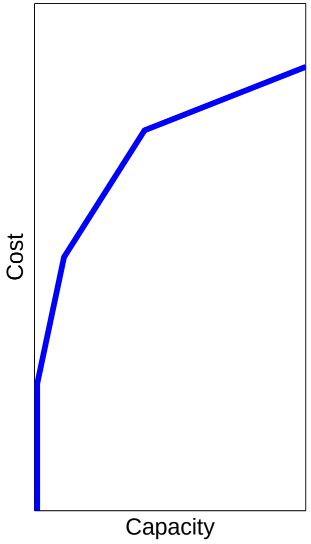
⇒ Uncapacitated fixed cost: dominant construction

Cost Models

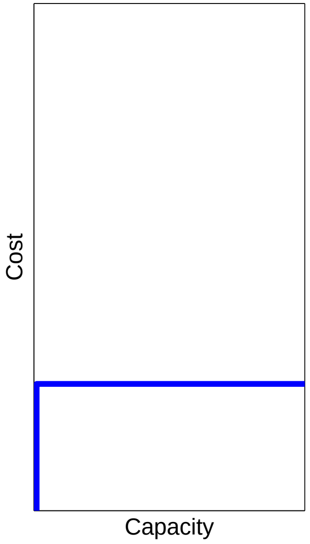
As a function of distance: affine or convex

As a function of link capacity: concave or constant

(a) General Concave Cost Model



(b) Uncapacitated Fixed Cost



A World of Graph Theory Problems

- A taxonomy of 16 problems

4 dimensions of variations, 2 possibilities each

- Some are difficult (NP-hard) and some are under-explored
- Two case studies shown here

Budget-Constrained Revenue Maximization

Uncapacitated fixed cost model, no existing tree, multiple levels

NP-hard

maximize $\sum_{v \in S} h_v r_v$ (total weighted survivability)

subject to $\sum_{(v,i) \in \widehat{E}} f^v_{v,i} \geq r_v + 1$ (number of disjoint paths)

$\sum_{(i,j) \in \widehat{E}} f^v_{i,j} = \sum_{(j,k) \in \widehat{E}} f^v_{j,k}$ (intermediate flow conservation)

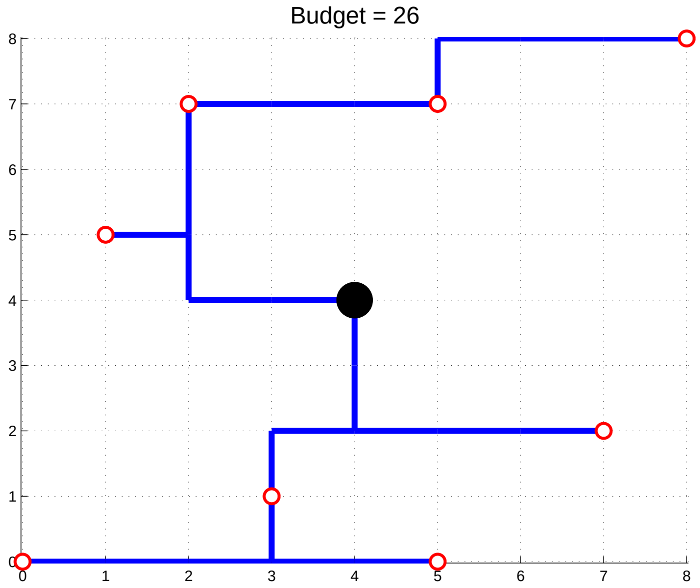
$x_e \geq f^v_{i,j}$ (*e*: undirected edge of *(i, j)*)

$B \geq \sum_{e \in E} c_e x_e$ (budget constraint)

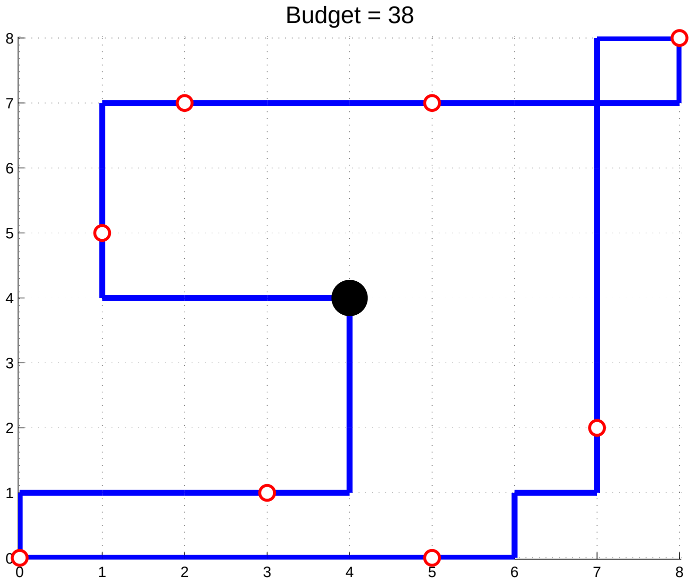
variables $r_v, f^v_{i,j} \geq 0, x_e \in \{0, 1\}$ (survivability, flow, edge selection)

Numerical Example

Min cost, No survivability
(Cost=26, Survivability=0)



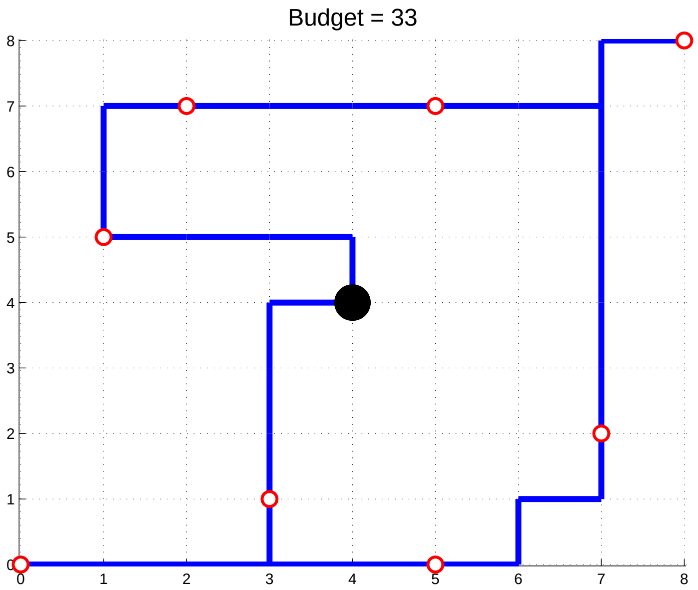
Min cost, Full survivability
(Cost=38, Survivability=8)



Numerical Example

Max-revenue access network with limited budget

Partial survivability (Cost=33, Survivability=6)



Provisioning Survivability for Existing Single-level Tree

- Min-cost incremental topology design to provide full survivability

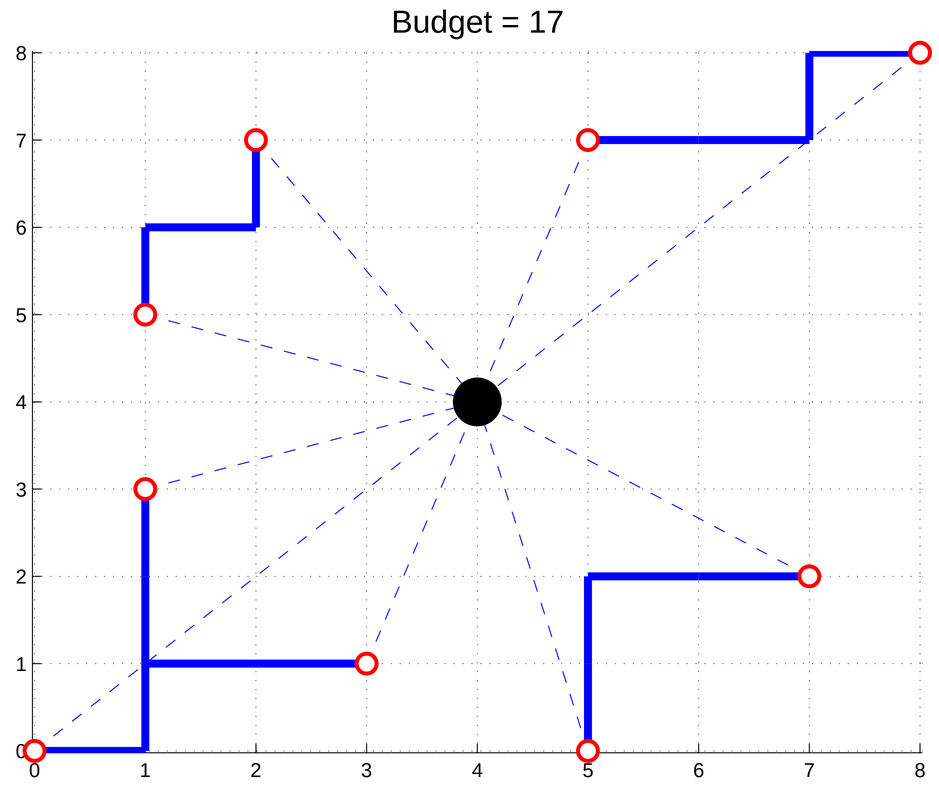
- Uncapacitated fixed cost model, tree exists, single level

- Equivalent to Terminal Backup problem: given (required) terminals, Steiner (optional) vertices, and weighted edges, find the cheapest subgraph where every terminal is connected to at least one other terminal (for backup purpose)

- Polynomial time

Numerical Example

Established tree in dots, optimized addition of backup links in solid



Time

Taxonomy of Problems

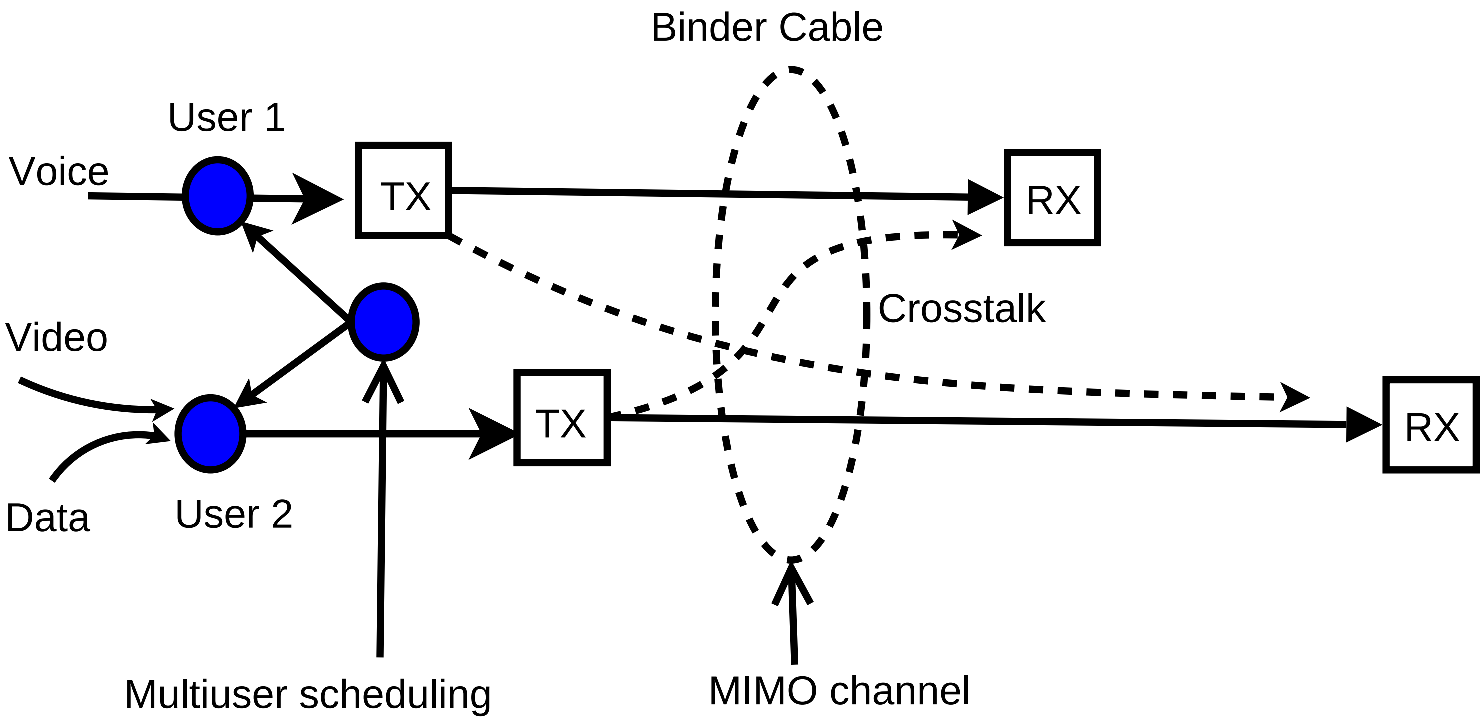
Question: Which point on rate region boundary to strike at?

QoS requirement	Characteristic	Control mechanism
Average throughput	Statistical	Multiuser scheduler
Average delay	Statistical	Multiuser scheduler
Hard delay bound	Deterministic	Priority queueing
Packet loss	Statistical	Priority queueing & Adm. Ctrl.
Inter-user fairness	Deterministic	Multiuser scheduler & Adm. Ctrl.

- Multiuser scheduling provides guarantees at the inter-user level
- Priority queueing provides guarantees at the intra-user level

Multiuser Revenue Based Scheduling

Multiuser scheduling in MIMO channel with different QoS characteristics

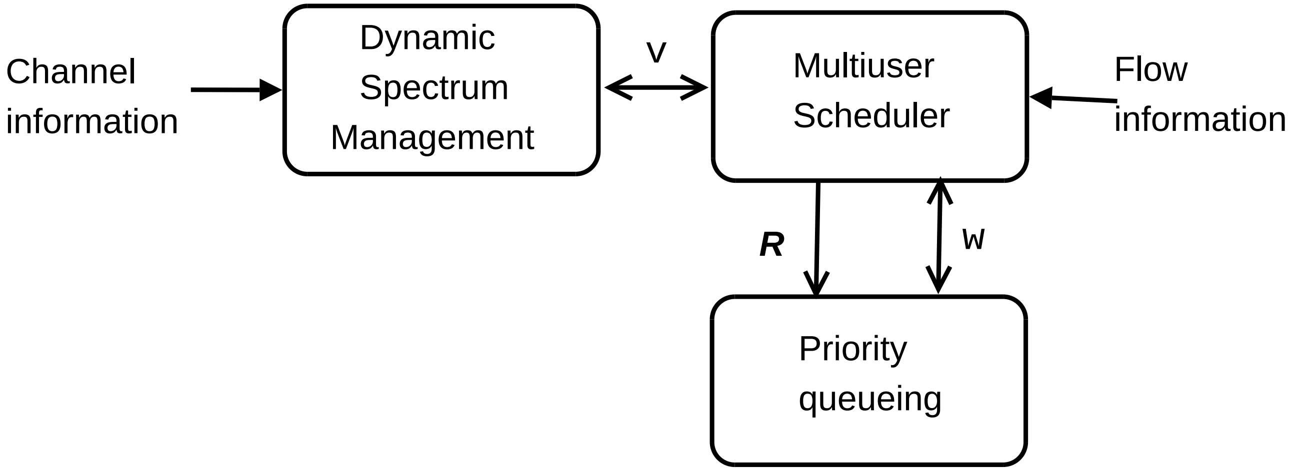


Example of Problem Formulation

$$
\begin{array}{rll}
\text{maximize} & R & \text{(Flow base rate)}\\
\text{subject to} & \beta_k R \le R_k(\mathbf{s}) x_k,\ \ k = 1, \ldots, K, & \text{(Rate constraint)}\\
 & \displaystyle\sum_{k=1}^{K} x_k \le 1, & \text{(Time share constraint)}\\
 & \displaystyle\sum_{n=1}^{N} s_k(n) \le P_k,\ \ k = 1, \ldots, K, & \text{(Power constraint)}\\
 & x_k \ge 0,\ \ k = 1, \ldots, K, & \\
 & s_k(n) \ge 0,\ \ k = 1, \ldots, K,\ \ n = 1, \ldots, N, & \\
\text{variables:} & R,\ \ x_k,\ \ \mathbf{s}_k, k = 1, \ldots, K. &
\end{array}
\tag{1}
$$

Joint Time Frequency Scheduling Algorithm

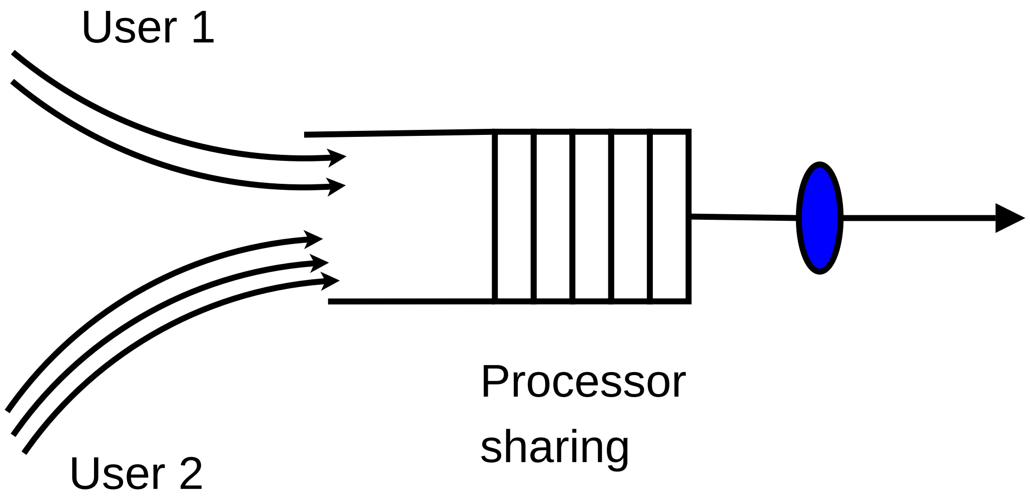
- Multiuser scheduling with central coordination



- Transmission of flows in time subproblem
- Dynamic spectrum management in frequency subproblem

Using Processor Sharing Model

- The PS model serves as a theoretical benchmark for stochastic performance metrics such as average delay
- A larger revenue corresponds to a larger flow throughput for each user



- Priority queueing differentiates application traffic flows in each user

Summary

Conclusion and Future Work

All three things at the same time:

- Presents intellectually challenging research issues in broadband access networking
- Motivates many new and difficult problems in optimization theory, information theory, signal processing, networking, graph theory, stochastic systems
- Offers an opportunity to make visible, tangible impacts to practical deployment

Next step: Empirical data verification of network algorithms

Next step: More solutions to this array of research problems

The Promise of FAST Copper Broadband Access

- Rate: Fast
- Reach: Ubiquitous
- Reliability: Survivable
- Quality: QoS for triple play

Contacts

chiangm@princeton.edu
www.princeton.edu/∼chiangm