FAST Copper For Broadband Access

Mung Chiang[1], Jianwei Huang[1], Dahai Xu[1], Yung Yi[1], Chee Wei Tan[1], Raphael Cendrillon[2]
[1] Electrical Engineering Department, Princeton University [2] Marvell Semiconductors

***Abstract*— This is an overview of the ongoing FAST Copper project, which is aimed at substantial improvements in rate, reach, reliability, and quality in copper-last-mile broadband access through fiber/DSL deployment, engineering innovations, and fundamental research. The project is funded by NSF, and is currently pursued jointly by Princeton University, Stanford University, and Fraser Research Lab. In this article, we outline the motivations, challenges, and research issues associated with the project, including connections with many branches of fundamental research, collaboration with industry partners, and emphasis on architectural decisions in broadband access networks. We also report some of the recent results by the Princeton team in each of the four dimensions: Frequency, Amplitude, Space, and Time.**

Keywords: Broadband access networks, Digital Subscriber Loop, Graph theory, Interference channel, Multi-carrier systems, Optic fiber communications, Optimization, Scheduling, Spectrum management, Statistical multiplexing, Topology design.

I. Introduction to FAST Copper

FAST Copper is a multi-year, U.S. NSF funded project that started in 2004, and is currently pursued jointly by the research groups of Mung Chiang at Princeton University, John Cioffi at Stanford University, and Alexader Fraser at Fraser Research Lab, and in collaboration with several industrial partners including AT&T. The goal of the FAST Copper Project is to substantially improve the rate, reach, reliability, and quality in copper-last-mile broadband access to everyone with a phone line, through the combination of fiber/DSL deployment, engineering innovations, and fundamental research.

This goal will be achieved through two threads of research: dynamic and joint optimization of resources in Frequency, Amplitude, Space, and Time (thus the name 'FAST') [1] to overcome the attenuation and crosstalk bottlenecks, and the integration of communication, networking, computation, modeling, and distributed information management for architectural design of broadband access networks.

Access networks are often the rate-reach-reliability-quality bottleneck of end-to-end connections in wide area networks. Realizing the vision of truly broadband and ubiquitous access to almost everyone in the U.S. is a formidable task, with many significant technical and socio-economic challenges. Although the fiber-to-the-home solutions promise to provide broadband delivery, the *labor costs* associated with fiber installation need to be divided over the number of customers served by the fiber. Such cost becomes increasingly expensive as the number of customers served decreases, which happens when fiber gets closer and closer to the customer, especially in suburban areas. That last segment labor cost of deployment is the dominant economic limitation in broadband access, especially given the population density in established suburban neighborhoods in U.S.

[1]FAST Copper project is completely different from the TCP FAST project at Caltech, which is a project that improves the transport layer protocol.

We propose to leverage the installed copper plant, which is by far the most ubiquitous access network in the U.S. The overall solution is a hybrid fiber/DSL deployment where fiber is pushed into the access network but copper takes over the last mile, thereby utilizing the best of ubiquity, broadband, reliability, and economic viability. Can *substantially higher* data rate and application throughput be attained over DSL through research innovations? We believe the answer is definitely positive. To achieve data rates significantly higher than the current levels on low-twist unshielded telephone wires demands thinking about transmission on copper wires in a new way. This project combines innovative optimization and signal processing techniques with novel network architectures and protocols, as well as an integrated plane of real-time control, computation, data collection, and auto-configuration, to enable an access infrastructure that is *both* broadband and ubiquitous.

After surveying the key ideas in FAST Copper in Section II, we will outline the challenges of architectural design for broadband access networks in Section III, and then provides a brief summary of some of the latest developments in 2005-2006 for Princeton's part of this actively ongoing project. This summary is presented along the dimensions of Frequency, Time, Amplitude, and Space, with more mature results for the Frequency axis of the project. We expect continuous progress to be made by all institutions in the project in the coming years.

II. Key Ideas and Technical Challenges

Traditionally, DSL broadband access networks have been analyzed by viewing each twisted pair as a separate communication channel, independent of other twisted pairs in the same binder cable, with a fixed pipesize supporting circuit-switched voice traffic. The key to realizing the vision of ubiquitous, readily deployable, and truly broadband access networks is to dynamically optimize the resources in the dimensions of Frequency, Amplitude, Space, and Time, in the multiple-input-multiple-output communication environment of DSL across multiple layers in the protocol stack.

Therefore, there are two shifts of mentality that underlines the wide range of activities in FAST Copper:

- The first key idea is that, instead of holding the traditional view that each twisted pair is an independent channel, we model a bundled cable of twisted pairs as one aggregate *multi-user* communication system. Multiple users *compete* against and *cooperate* with each other in this system. We can explicitly take into account the *crosstalk* effects (both near-end and far-end) that currently form the data rate bottleneck, and to exploit potential *cooperation* in sharing limited resources

- The second key idea is that today's traffic over broadband access, including voice, data, and video, are predominately supported by packet switched IP. We can exploit the *burstiness* of the application traffic through aggressive statistical multiplexing, with admission control, traffic shaping, scheduling, and priority queuing mechanisms to ensure the desired tradeoff between the number of application flows supported and the Quality of Service (QoS) attainable.

There are two major bottlenecks to DSL broadband access today: attenuation and crosstalk. We will see solutions from the "Space" dimension of the project to tackle the problem of attenuation, and solutions from the "Frequency", "Amplitude", and "Time" dimensions to tackle the problem of crosstalk. Note that we have not even brought in factors such as wider bandwidth and multiple twisted-pairs.

- *Frequency*. In the physical layer, new techniques can be developed based on improving spectral utilization, mitigating multi-user interference, and exploiting multi-user cooperation. Through dynamic adaptation and utilization of frequency spectrum, such as power control, bit loading, or vectored transmission, Dynamic Spectrum Management (DSM) [26] allows maximum flexibility in allocating rates among competing flows, achieves much higher total data rates, and extends the reach of broadband access.
- *Time*. FAST Copper also leverages the potential for time division multiplexing based on the application layer burstiness of data traffic from and to the end hosts. In most communication-theoretic investigations, it is assumed that there is always an infinite backlog of bits that need to be transmitted per user, thus taking out the latency considerations and the temporal dimension. By jointly considering the application layers, burstiness of the required bandwidth provides another degree of flexibility of statistical multiplexing along the temporal axis.
- *Space*. When building robust and efficient broadband access networks, two issues are particularly important: how can a hybrid fiber/twisted pair architecture be designed to utilize the best of fiber-based and copper-based communication potentials, and how can a logical topology be designed to offer fast-recovery after natural failures or malicious attacks?
- *Amplitude*. We propose to install active 'amplitude control' mechanisms to shape the flow intensities at the edge to provide different QoS classes through admission control and dynamic bandwidth allocation. At the same time, a network management system constantly probes, measures and monitors the cable and its environments, receives data rate requests from user terminals, and periodically shapes the rate each user is allowed to transmit and receive per time frame.

In summary, by modeling the whole binder of copper wires as one multi-carrier interference channel, with resources ranging from the physical layer to the application layer, we can dynamically optimize over Frequency, Amplitude, Time, and Space, in a stable, robust, and complementary way. Collectively, these four degrees of freedom offer many exciting opportunities to make tangible practical impacts.

At the same time, progress in the project come from solving important problems in the fundamental research disciplines of information theory (multi-carrier interference channel), signal processing (multi-user transceiver design), optimization theory (nonconvex and coupled problems), graph theory (survivable tree topology design), stochastic theory (processor sharing and queuing networks), distributed control (feedback control at different timescales), and network protocol design (resource allocation and functionality allocation).

III. Architectural Issues in Broadband Access

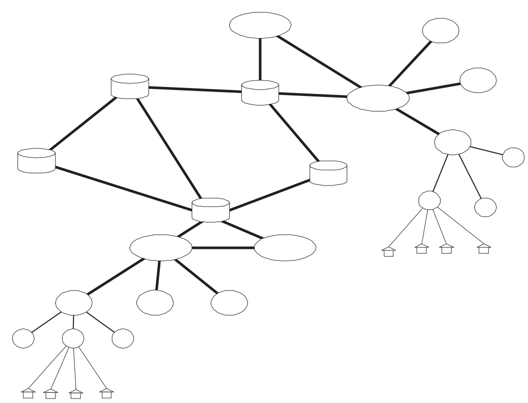


Fig. 1. Horizontal decomposition.

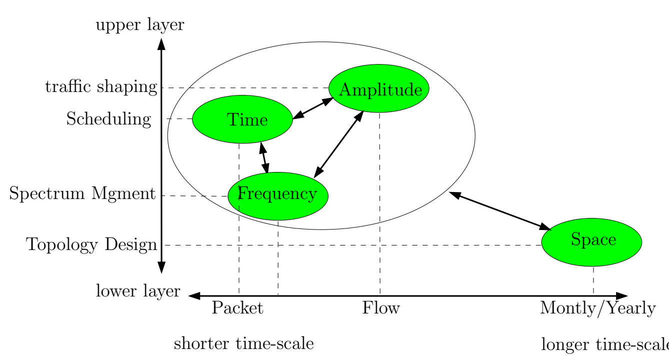


Fig. 2. Vertical decomposition and timescale separation.

A. Functionality Allocation

Increasing data rate by 10 times (or more) over twisted-pair already presents tremendous technical challenges. We need to significantly improve both the digital signal processing algorithms in the physical layer and the architecture/protocol design methodologies in the "upper" layers. Even more challenging is the need to carefully investigate the coupling effects across multiple modules and across network elements, so that end-user experience over broadband access networks is enhanced. Indeed, one of the most important aspect of FAST Copper, or any broadband access network design, is on the access network *architecture*. Following the notion of "architecture first", we open the technical discussion of the paper with this section on architectural choices.

Architecture here refers to *functionality allocation*: which functional module and network element does what, and how to connect them. Functionality allocation is often more influential, harder to change, and less quantitatively understood than any specific resource allocation scheme. Metrics of measuring the pros and cons of designs of functionality allocation often remain fuzzy today, and are drawn from a combination of performance metrics, cost and complexity metrics, and Network X-ities (e.g., evolvability, scalability, manageability, diagnosability, optimizability) metrics. Recent results in "Layering as Optimization Decomposition" [12] have offered a useful framework for layered network architecture.

There are unique challenges in architectural issues in broadband access networks, which are outlined in the next subsection before a case study discussed further in the following subsection.

B. Horizontal and Vertical Decompositions

There are two types of functionality allocations:

- First, we use the term "horizontal decomposition" to refer to the geographic distribution of control into various network elements, e.g., in Figure 1, from households to remote terminals and central offices (represented by circles), to larger central offices (represented by ovals), and to backbone acquisition, distribution, and video servers (represented by cylinders).
- Second, we use the term "vertical decomposition" to refer to the modularized design into a protocol stack. Ideally for performance optimization, a fully integrated and joint design would be best. However, for many reasons such as evolvability and manageability, modularized design is necessary.

Horizontally and vertically decomposable modules has a variety of coupling relationships, depending on their timescales and target service models. As an example of such couplings, consider the functionality of error control and recovery. We can choose from the following: hop-by-hop forward error correction or feedback based Automatic Repeat reQuest, or end-to-end connectionless resilient-UDP or connection-oriented TCP, or coding over packets at the application level. Which combination to choose from involves both vertical decomposition (e.g., physical layer or application layer) and horizontal decomposition (e.g., hop-by-hop or end-to-end).

In the horizontal decomposition, the problem of where to place video servers involves the tradeoff between response time and scalability, closer to the customers, faster the response but lower the scalability. Similarly, the problem of where to place distribution servers defines the boundaries of multicast groups. An even larger issue is on "how big should the access networks be". The answer depends on the tradeoff among a variety of factors, from reliability of access tree and feasibility of big switches to complexity of backbone network and ease of network management.

In the vertical decomposition, the four dimensions of F, A, S, and T are all coupled, sometimes in unexpected ways. As a thought experiment in the extreme, we can think of each of these dimensions as capable of tackling crosstalk: through spatial division multiplexing, time division multiplexing, frequency division multiplexing, and rejection of all flows that would transmit at the same time. Obviously, these are extreme measures, and tradeoffs are necessary in any design. But they do highlight the fact that these four dimensions are not uncorrelated degrees of freedom. More specifically, as shown in Figure 2,

- Topology design (e.g., placement of various servers and schedulers) determines feasibility of control in the time and amplitude dimensions, since excessive propagation delay due to geographical distance may lead to instability of scheduling at the packet level and impossibility of admission control of short flows.
- Topology design also determines the crosstalk channel gains, which are the parameters to spectrum management algorithms in Frequency dimension and limit the best rate regions attainable.
- Time and frequency are clearly coupled design freedoms. Each spectrum management or scheduling can mitigate crosstalk. As will be shown, advanced versions of dynamic spectrum management provides a convex rate region, thus rendering the simple time division scheduling inferior. The timescale of spectrum management depends on the speed of convergence (time complexity) of spectrum management algorithms, and that of scheduling depends on the granularity of schedules: anywhere from packet level schedules to flow level ones. Furthermore, the capability in the time dimension depends on the properties of spectrum management in the frequency dimension.
- Admission control in the amplitude dimension clearly depends on the rate regions that are assumed to be attainable by the underlying spectrum management and scheduling algorithms.
- Interactions between the amplitude and time dimensions will be further explained in the next subsection.

It is important to realize that the timescales of different functionalities can vary significantly. A key point is that more explicit time-scale separations often enables easier decompositions by lowering the "price of modularization". For example, as shown in Figure 2, problems such as topology design is dealt with on a monthly or yearly basis, whereas time and amplitude-related problems is tackled at the much faster timescale of packet-level or flow-level dynamics. Thus, it does not seem to be of great significance to explicitly consider the space domain in the time and amplitude domain. This is due

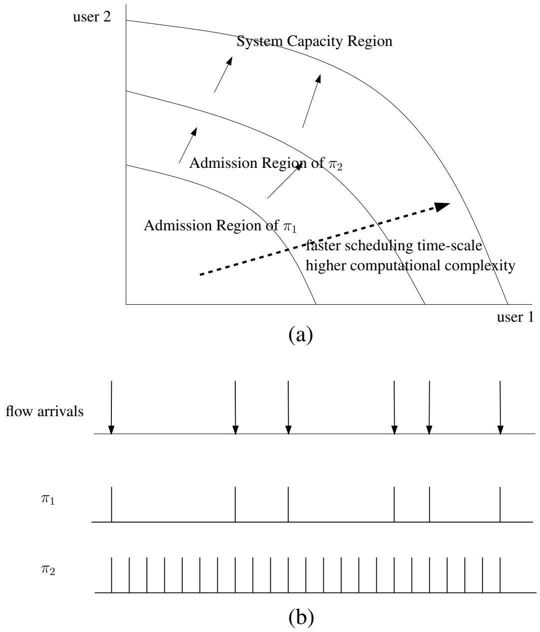


Fig. 3. Interdependence of statistical multiplexing, scheduling, and spectrum management.

to the fact that time-scale separation between two dimensions leads to (i) quasi-stationary regime (i.e., nearly constant) and (ii) fluid regime (i.e., nearly average), from the perspective of the dimensions operating with faster time-scale and slower time-scale, respectively.

C. A Case Study: Coupling between Frequency and Time

In this subsection, we discuss the coupling between frequency and time domain, which further highlights some of the issues and tradeoffs involved in architectural design of broadband access networks.

As an illustrative example, consider two scheduling algorithms (in the time dimension), π_1 and π_2, where π_1 and π_2 are supported by two different DSM algorithms operating at the flow-level and packet-level timescales, respectively (see Figure 3). We denote by $\mathcal{C}$ the rate region, obtained by DSM in the frequency dimension. There are two users u_1, u_2 and their set of QoS requirements (e.g., packet loss probability) are denoted by Q^1 and Q^2. We also define $A(\pi)$ to be the admission region of scheduling π, i.e., the set of (long-term) average arrival vectors (λ^1, λ^2), such that their QoS are satisfied under the scheduling π. In this setup, our objective is to enlarge the admission region[2], and the scheduling π_1 (or π_2) in the time dimension is to schedule rates for two users at flow-level (or packet-level) time-scale.

Assuming that the time-scale of flow-level arrivals and departures is sufficiently slow, such that the stationary behavior can be seen during the inter-arrivals of flows, one potential tool of π_1 to schedule the rates and decide its admission region is the mechanism of effective bandwidth (either in the sense of large buffer asymptote or many source asymptote [14], [19], [30]). On the other hand, the π_2 is able to schedule the arbitrary rates (which is inside the rate region) due to the support of the time-scale of the underlying DSM algorithm. Full support of a DSL algorithm in the frequency at the fast time-scale enables the π_2 to use inherent "opportunism", i.e., allocating more rates dynamically to the user whose transient situation requires prioritized service (e.g., the user with longer queue length) [2], [5], [25].

This implies that $\mathcal{A}(\pi_1) < \mathcal{A}(\pi_2) < \mathcal{C}$. Although π_1 with time-scale separation between time and frequency leads to better decomposability, it has to pay the price of decreasing system performance (i.e., decreasing admission region). This again illustrates the fundamental tradeoff between modularity and performance. It is important to have an efficient DSM algorithm which fills the gap, and to develop a joint statistical multiplexing, scheduling, and DSM algorithm to achieve better system performance, i.e., a low-complexity scheduling algorithm π, such that $\mathcal{A}(\pi) \approx \mathcal{C}$. This motivates our work in Sections IV and VII.

In concluding this section on architectural choices, we make an observation that often the slower timescale functionality may render the faster timescale functionality useless, but not the other way around. In the case study of amplitude-time coupling, a complicated, opportunistic scheduling algorithm becomes useless if admission control is conservative and admits only flows that can be scheduled with simple methods. On the other hand, no matter what kind of scheduling or DSM methods are used, admission control is required to at fulfill the basic functionality of rejecting flows that cannot be supported.

After demonstrating that there are many architectural issues in broadband access networking and a quantified study of such issues form one of the most important research challenges, we now move on to summarize the latest results on each of the individual dimensions in FAST Copper project.

IV. Frequency: Spectrum Management

A. Overview

Competition among users in a multi-carrier interference channel like DSL can be mitigated in the Frequency domain either by dynamic spectrum management (DSM) or by bonded transceiver design [26]. We focus on DSM solutions in this section, which do not require new transceiver chip-sets.

The representative DSM algorithms in the literature include *Iterative Water-filling* (IW) algorithm [31], *Optimal Spectrum Balancing* (OSB) algorithm [10] and *Iterative Spectrum Balancing* (ISB) algorithm [9], [22]. IW algorithm is a completely autonomous algorithm with a linear complexity in the number of users. In IW, each line maximizes its own data rate by waterfilling over the noise and interference from other lines. Due to its selfish nature, IW leads to a highly-suboptimal performance in the widely-encountered near-far scenarios, such as mixed central office and remote terminal deployments of ADSL and upstream VDSL (see the example in Section IV-B). Both OSB and ISB are centralized algorithms. The OSB algorithm achieves the optimal performance by using

[2]This objective is an indirectly equivalent representation of maximizing injected users in the system or maximizing the system provider's revenue.

TABLE I

COMPARISON OF VARIOUS DSM ALGORITHMS

Algorithm	Operation	Complexity	Performance	Reference
IW	Autonomous	$O(KN)$	Sub-optimal	[31]
OSB	Centralized	$O(Ke^N)$	Optimal	[10]
ISB	Centralized	$O(KN^2)$	Near optimal	[9], [22]
ASB	Autonomous	$O(KN)$	Near optimal	[16]

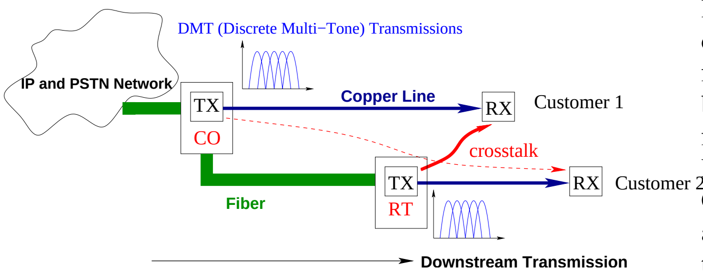


Fig. 4. The typical mixed CO/RT deployment (near-far scenario) for downstream transmission in an ADSL network. The CO (Central Office) is connected to the IP and PSTN Network via fiber; the RT (Remote Terminal) is connected to the CO via fiber. The CO and RT terminate at end customer homes through copper twisted-pair lines (telephone lines), where data rate is limited by crosstalk.

a maximization of a weighted rate-sum across all users, which explicitly takes into account the damage done to the other lines when optimizing each line's spectra. Unfortunately OSB has an exponential complexity in the number of users, making it intractable for DSL network with more than 5 lines. ISB improves over OSB algorithm by implementing the weighted-rate sum optimization in an iterative fashion over the users. This leads to a quadratic complexity in the number of users. But it still requires centralized computation, which is often impossible since interfering users may not under one controller in the CO.

In summary, no previous DSM algorithms can provide both low complexity, autonomous operation and near-optimal rate region. Our recently proposed *Autonomous Spectrum Balancing (ASB)* [16] algorithm overcomes the problem. The comparison of various DSM algorithms in terms of performance and complexity is given in Table I.

The IW, OSB and ISB mentioned above all assume synchronous transmissions of the modems, which allows crosstalk to be modeled independently on each tone. In practice, the signal transmitted on a particular tone of one modem will appear as crosstalk on a broad range of tones on the other modems. This inter-carrier-interference (ICI) significantly complicates the DSM problem further. Two centralized DSM algorithms that deal with ICI have been proposed in [11]. We have proposed a variation of the ASB algorithm [8] to tackle the ICI in a distributed fashion. Due to space limitations, we focus the discussions on the synchronous transmission case here.

B. Network Model

Discussions in this section hold for any DSL systems topology. To be concrete, we will often examine the typical near-far deployment scenario for downstream ADSL transmissions, as shown in Fig. 4. There are two twisted-pair copper lines in the network. The first line is from the Central Office (CO) to customer 1. Since customer 2 is far away from CO, the service provider deploys a Remote Terminal (RT) near the edge of the network, which connects with customer 2 through a relatively short copper line. In the downstream transmission case shown in the figure, the transmitting modems (TX) are located at the CO and RT, and the receivers (RX) are at the customer homes. Each DSL modem transmits over multiple frequency tones (carriers). Multiple lines sharing the same binder generate crosstalks (interferences) to each other on all frequency tones. Although RT extends the footprint of the DSL network, it also generates excessive interference to the CO line due to the physical proximity between the RT TX and the CO RX. However, CO TX generates little crosstalk to RT RX due to the long distance between them. This near-far problem is very typical in the DSL deployment in the U.S., and has become the major performance bottleneck.

Next we formally introduce the mathematical system models. Consider a DSL network with a set $\mathcal{N} = \{1, ..., N\}$ users (i.e., lines, modems) and $\mathcal{K} = \{1, ..., K\}$ tones (i.e., frequency carriers). Assuming the standard *synchronous* discrete multi-tone (DMT) modulation is applied, transmission can be modeled independently on each tone k as $\boldsymbol{y}^k = \boldsymbol{H}^k \boldsymbol{x}^k + \boldsymbol{z}^k$. The vector $\boldsymbol{x}^k \triangleq \{x_n^k, n \in \mathcal{N}\}$ contains transmitted signals on tone k, where x_n^k is the signal transmitted by user n at tone k. Vectors $\boldsymbol{y}^k$ and $\boldsymbol{z}^k$ have similar structures: $\boldsymbol{y}^k$ is the vector of received signals on tone k, and $\boldsymbol{z}^k$ is the vector of additive noise on tone k and contains thermal noise, alien crosstalk and radio frequency interference. We denote the channel gain from transmitter m to receiver n on tone k as $h_{n,m}^k$. We denote the transmit power spectrum density (PSD) $s_n^k \triangleq \mathcal{E}\left\{\left|x_n^k\right|^2\right\}$, where $\mathcal{E}\{\cdot\}$ denotes expected value. The vector containing the PSD of user n on all tones as $\boldsymbol{s}_n \triangleq \{s_n^k, k \in \mathcal{K}\}$.

Assume that each user treats interference from other modems as noise. When the number of interfering users is large, the interference can be well approximated by a Gaussian distribution. Under this assumption the achievable bit rate of user n on tone k is

$$b_n^k \triangleq \log\left(1 + \frac{1}{\Gamma} \frac{s_n^k}{\sum_{m \neq n} \alpha_{n,m}^k s_m^k + \sigma_n^k}\right), \tag{1}$$

where $\alpha_{n,m}^k = \left|h_{n,m}^k\right|^2 / \left|h_{n,n}^k\right|^2$ is the normalized crosstalk channel gain (with $\alpha_{n,n}^k \triangleq 0, \forall k, n$), and σ_n^k is the noise power density normalized by the direct channel gain $\left|h_{n,n}^k\right|^2$. Here Γ denotes the SINR-gap to capacity, which is a function of the desired BER, coding gain and noise margin [27]. For notational simplicity, we absorb Γ into the definition of $\alpha_{n,m}^k$ and σ_n^k (i.e., let $\Gamma = 1$). The bandwidth of each tone is normalized to 1. Each user n is typically subject to a total power constraint P_n, due to the limitations on each modem's analog frontend: $\sum_{k \in \mathcal{K}} s_n^k \leq P_n$. The data rate on line n is thus $R_n = \sum_{k \in \mathcal{K}} b_n^k$.

The spectrum management problem is defined as follows

$$\max_{\{\boldsymbol{s}_n \geq \mathbf{0}, n\in\mathcal{N}\}} \sum_n w_n R_n \text{ s.t. } \sum_{k\in\mathcal{K}} s_n^k \leq P_n, \forall n. \tag{2}$$

where w_n is a nonnegative weight coefficient of user n. It is known [10] that in the asymptote of large number of tones, scalarization technique (varying the weights $\{w_n\}$) traces out the entire Pareto-optimal boundary of the rate region. Due to interference between lines, Problem (2) is nonconvex. Furthermore, it is highly coupled across lines (due to crosstalk) and tones (due to total power constraint), making it a very difficult optimization to solve. In particular, any algorithm that globally solves (2) must have knowledge of all crosstalk channels and background noise spectra, forcing it to operate in a centralized fashion. In order to overcome this difficulty, we observe that for optimal solutions of (2) each user adopts a PSD that achieves a fair compromise between maximizing their own data-rate and minimizing the damage they do to other lines within the network.

Based on this insight, we introduce the concept of a "reference line", a virtual line that represents a "typical" victim within the DSL system. Since network operators are typically concerned with maximizing the rate achieved by the worst line within their network, the reference line typically corresponds to the longest line in the network (e.g. the CO distributed line in a mixed CO/RT scenario, such as that in Section IV-E), which has the weakest direct channel and receives relatively stronger crosstalk from other users. Then instead of solving (2) , each user tries to maximize a weighted sum of its own rate and the reference line's rate, subject to its own rate and total power constraints.

Since the main purpose of introducing the reference line is to characterize the "damage" that each user does to the network, we will make the achievable rate of the reference line user dependent. In other words, *from user n's point of view*, the reference line's rate is $R_{n,ref} \triangleq \sum_{k\in\mathcal{K}} \tilde{b}_n^k$, where the achievable bit rate on tone k is defined as

$$\tilde{b}_n^k \triangleq \log\left(1 + \frac{\tilde{s}^k}{\tilde{\alpha}_n^k s_n^k + \tilde{\sigma}^k}\right), \tag{3}$$

The coefficients $\left\{\tilde{s}^k, \tilde{\sigma}^k, \tilde{\alpha}_n^k, \forall k, n\right\}$ are parameters of the reference line and can be obtained from long-term field measurements[3]. Since the crosstalk channel can be regarded as time-invariant in the DSL wireline network, the parameters of the reference lines are known to users a priori through channel measurements. Intuitively, the reference line serves a penalty term in each user's optimization problem to avoid purely selfish behavior, and eliminates the need of explicit message passing amongst users.

Therefore, instead of solving Problem (2), which requires global information, we let each user n solve the following problem in ASB algorithm (treating the crosstalks from other users as fixed Gaussian noise):

$$\max_{\boldsymbol{s}_n \geq \mathbf{0}} w_n R_n + R_{n,ref} \quad \text{s.t. } \sum_{k\in\mathcal{K}} s_n^k \leq P_n. \tag{4}$$

In other words, we let each user solve a problem locally, treating the reference line as a "static pricing" term. Users then iterate until PSD converges. While static pricing is used for static coupling across the users to avoid practically infeasible centralized computation or message passing, dynamic pricing can still be used for dynamic coupling across tones for each individual user.

C. ASB Algorithm

In the ASB algorithm, each user n solves Problem (4) using a dual-based decomposition algorithm. First, incorporating the total power constraint into the objective function of (4) leads to the following Lagrangian,

$$L_n \triangleq w_n R_n + R_{n,ref} - \lambda_n \sum_{k\in\mathcal{K}} s_n^k, \tag{5}$$

where λ_n is the dual variable of user n and needs be chosen such that $\sum_k s_n^k = P_n$ or $\lambda_n = 0$ (i.e., complementary slackness). The Lagrangian can also be written as $L_n = \sum_k L_n^k$, where

$$L_n^k = w_n b_n^k + \tilde{b}_n^k - \lambda_n s_n^k. \tag{6}$$

The optimal PSD on each carrier k (given λ_n) is obtained by

$$s_n^{k*}(\lambda_n) = \arg\max_{s_n^k \in [0,P_n]} L_n^k\left(w_n, \lambda_n, s_n^k, s_{-n}^k\right), \tag{7}$$

where $s_{-n}^k = \left\{s_m^k, \forall m \neq n\right\}$. Although L_n^k is nonconvex, $s_n^{k*}(\lambda_n)$ can be found by solving the first order condition, $\partial L_n^k / \partial s_n^k = 0$, which has three analytical solutions. Based on the relationship between $\sum_k s_n^{k*}(\lambda_n)$ and P_n, user n adjusts the value of λ_n in an outer loop until the complementary slackness condition is satisfied. Although Problem (4) is nonconvex, we know from [10] that the corresponding duality gap of Problem (4) is asymptotically zero (when the number of tones becomes large), thus the dual-based approach leads to an optimal primal solution.

Each user solves it own version of Problem (4) in an distributed and iterative fashion until the PSDs of all users converge.

D. Convergence Analysis

The nonconvexity of L_n^k in (7) makes it difficult to prove the convergence of ASB algorithm in general. In the two-user case, we can show the following.

Theorem 1: Consider a two-user system with fixed $\boldsymbol{w}$ and $\boldsymbol{\lambda}$. There exists at least one fixed point of ASB, and the algorithm converges if users start from initial PSD values $\left(s_1^k, s_2^k\right) = (0, P_2)$ or $\left(s_1^k, s_2^k\right) = (P_1, 0)$ on all tones.

The proof of Theorem 1 uses supermodular game theory [28] and strategy transformation similar to [15], and is omitted here due to space limitation.

[3] In fact, the reference line concept is already used in existing VDSL standards such as T1.424-2004. Good choices for reference lines have been defined based on extensive studies. ASB shows how to use this intuitive concept in a stable and near-optimal manner

To reduce the computation complexity and gain more insights into the solution structure, we rewrite the reference line rate using linear approximation and high Signal-to-Noise Ratio (SNR) approximation, i.e.,

$$\tilde{b}_n^k\left(s_n^k\right) \approx \left(\log\left(\frac{\tilde{s}^k}{\tilde{\sigma}^k}\right) - \frac{\tilde{\alpha}_n^k s_n^k}{\tilde{\sigma}^k}\right)\mathbf{1}_{\{\tilde{s}^k>0\}}, \quad (8)$$

where $\mathbf{1}_{\{\mathfrak{A}\}}$ is the indictor function and equals to one when event $\mathfrak{A}$ is true. Under (8), Problem (4) becomes a convex optimization problem, and the corresponding optimal PSD can be found in close form as

$$s_n^{k*}\left(\lambda_n\right) = \left[\frac{w_n}{\lambda_n + \frac{\tilde{\alpha}_n^k}{\tilde{\sigma}^k}\mathbf{1}_{\{\tilde{s}^k>0\}}} - \sum_{m\neq n}\alpha_{n,m}^k s_m^k - \sigma_n^k\right]^+,$$

where $[x]^+ = \max\{x, 0\}$. This is a water-filling type of solution, with different water-filling levels for different tones. We name it *frequency selective waterfilling*. The solution is intuitively satisfying: the PSD for user n should be smaller when the power constraint is tighter (i.e., λ_n is larger), or the crosstalk channel to the reference line $\tilde{\alpha}_n^k$ is higher, or the noise level on the reference line $\tilde{\sigma}^k$ is smaller, or there is more interference plus noise $\sum_{m\neq n}\alpha_{n,m}^k s_m^k + \sigma_n^k$ on the current tone.

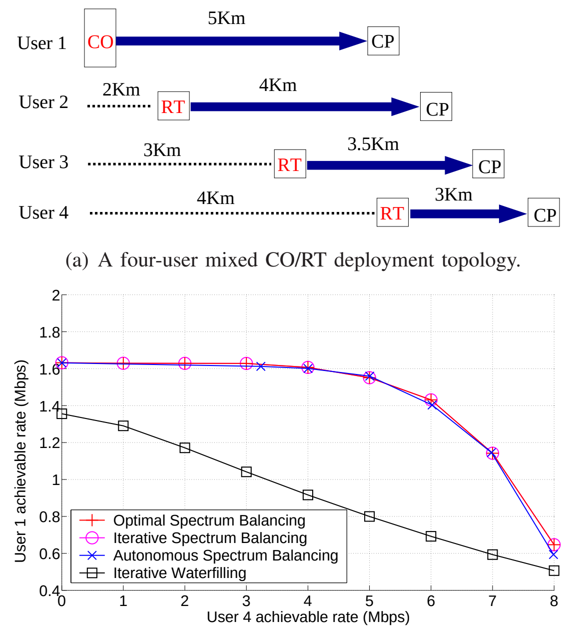
Stronger convergence results can be obtained with the approximation (8) for both sequential and parallel updates. Denote $s_n^{k,t}$ as the PSD of user n on tone k after iteration t, where $\sum_k s_n^{k,t} = P_n$ is satisfied at the end of any iteration t for any user n. In the sequential updates, only one user will change its PSD at any time. One iteration is defined as one round of updates of all users. In the more realistic but harder-to-analyze parallel updates, time is divided into slots, and the users update their PSDs simultaneously in each time slot, where the λ^n is adjusted such that the power constraint is tight.

Theorem 2: Assume $\max_{m\neq n,k}\alpha_{n,m}^k < \frac{1}{N-1}$, then the ASB algorithm under high SNR approximation globally and geometrically converges to the unique fixed point in an N-user system, with either sequential or parallel updates.

Theorem 2 recovers the convergence of iterative water-filling in an N-user case with sequential updates (proved in [13]) as a special case. Moreover, the convergence proof for the parallel updates turns out to be simpler than that for sequential updates.

E. Simulation Results

A four-user mixed CO/RT scenario is selected to demonstrate the performances of various DSM algorithms. As depicted in Fig. 5(a), user 1 is CO distributed, whilst the other three users are RT distributed. Due to the different distances among the corresponding transmitters and receivers, the RT lines generate strong interferences into the CO line, whilst experiencing very little crosstalk from the CO line. The target rates of users 2 and 3 have both been set to 2 Mbps (by adjusting the corresponding weights w_n). For a variety of different target rates of user 4, user 1 (the CO line) attempts to maximize its own data-rate either by transmitting at full power in IW, or by setting its corresponding weight w_{co} to a very large number in OSB, ISB and ASB. This produces the rate regions shown in Fig. 5(b), which shows that ASB achieves near optimal performance (almost the same as OSB and ISB), and significant gains over IW. For example, with a target rate of 1 Mbps on user 1, the rate on user 4 reaches 7.3 Mbps under ASB algorithm, which is a 121% increase compared with the 3.3 Mbps achieved by IW.



(a) A four-user mixed CO/RT deployment topology.

(b) Rate regions obtained by various DSM algorithms.

Fig. 5. Performance comparison of various dynamic spectrum management (DSM) algorithms.

V. Amplitude: Statistical Multiplexing

A. Overview

DSM as discussed in the last section assumes a static view of the application traffic. We can further exploit the burstiness the traffic. In this section, we consider statistical multiplexing at the time scale of flow level (e.g., "scheduling" policy π_1 in Section III). The goal here is to determine how much traffic to be admitted into the network such that resource is fully utilized subject to the QoS requirements.

Statistical multiplexing has been widely studied in wireline networks (e.g., [4], [21], [24] and the references therein), where the underlying link capacities are assumed to be fixed. In DSL network, however, the link capacities are closely coupled among users due to crosstalk interferences, and can be dynamically "shuffled" across users based on traffic demands. Statistical multiplexing at the link layer should explicitly take this into consideration. Therefore, we aim at joint optimization of various system resources (e.g., bandwidth, buffer) to achieve the best possible statistical multiplexing performance. We limit our discussions to the delay insensitive elastic traffic here for illustration purpose. For more detailed discussions including how to deal with real-time traffic, see [17].

B. An Example of Problem Formulation

For delay insensitive data traffic of user n, the QoS requirement is defined as that the packet loss probability due to buffer overflow should be less than a given tolerance threshold ϵ_n. This probability is determined by the traffic characteristics (i.e., mean, peak, variance) and the allocated resource to the traffic (i.e., bandwidth and buffer). When ϵ_n is very small and the allocated buffer is very large, the bandwidth requirements of users' data traffic can be estimated accurately using the concept of *effective bandwidth* [19].[4]

We assume that all flows of user n belong to the same class, which can be characterized by an average rate a_n, a peak rate r_n, and an effective bandwidth $\nu_n(\delta_n)$. Here δ_n is a parameter that increases with ϵ_n and decreases with allocated buffer size B_n. The specific function form of ν_n depends on the statistical nature of the flow of user n (for more details, see [19]). In general, we have $a_n \leq \nu_n(\delta_n) \leq r_n$, with $\nu_n(\delta_n) = a_n$ when δ_n approaches 0, and $\nu_n(\delta_n) = r_n$ when δ_n approaches ∞. This means that the effective bandwidth becomes closer to the average rate when the QoS requirement is not stringent and buffer size is large, and it will approach the peak rate when the QoS requirement becomes more stringent and the buffer size decreases. The specific function form of ν_n depends on the stochastic nature of user n's traffic. Further refinements of the effective bandwidth concepts can be found in [14], [30], but here we will stick with the definitions in [19] for the purpose of illustration.

Our goal is to determine the admission region of the network, i.e., how many flows of each user can be admitted by the network, $\boldsymbol{g} = \{g_n, n \in \mathcal{N}\}$. In other words, the average rate achieved by user n would be $g_n a_n$. This will depend on the weighted of the users, i.e., w_n for user n. By adjusting the weights, we can trace out the entire boundary of the admission rate region.

In the case where all users have the same QoS requirements, i.e., $\epsilon_n = \epsilon$ for all n, we can let all users share the total buffer with size B. In the more general case where users have different ϵ_i, we may need to allocate a separate queue for each user n with different buffer size B_n with $\sum_n B_n = B$. In that case, the effective bandwidth of user n can be written as $\nu_n(\epsilon_n, B_n)$. Denote the rate of DSL link n (user n) as c_n, then the total effective bandwidth of user n's traffic should satisfy $g_n \nu_n(\epsilon_n, B_n) \leq c_n$. The value of vector $\boldsymbol{c} = \{c_n, n \in \mathcal{N}\}$ needs to be chosen from the feasible rate region, $\mathcal{C}$, which is determined by the crosstalk channel gains at the particular scheduling time instant and the solutions of the specific spectrum management algorithm. Mathematically, we have the following problem formulation:

[4]Here bandwidth refers to the transmission rate, instead of frequency band in Section IV.

$$\begin{aligned} \text{maximize} \quad & \sum_n w_n a_n g_n \\ \text{subject to} \quad & g_n \nu_n(\epsilon_n, B_n) \leq c_n, \forall n \\ & \sum_n B_n = B, \\ & \boldsymbol{c} \in \mathcal{C}, \\ \text{variables} \quad & \boldsymbol{g}, \boldsymbol{c}, \boldsymbol{B} \geq 0 \end{aligned} \tag{9}$$

where vector $\boldsymbol{B} = \{B_n, n \in \mathcal{N}\}$. Finding the global optimal solution of Problem (9) efficiently can be difficult since Problem (9) is typically non-convex. However, a feasible solution can nevertheless be computed with a two-stage iterative algorithm as described next.

C. An Example of Solution Algorithm

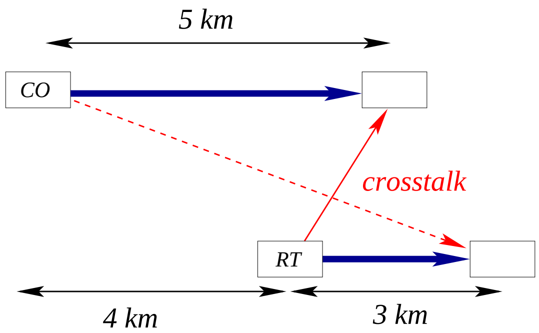
We propose a two-stage AM algorithm to solve Problem (9). Since each stage of the AM algorithm improves the objective of Problem (9), which is upperbounded, the AM algorithm always converges.

1) Stage 1 (bandwidth allocation): We fix the buffer allocations ($\boldsymbol{B}$) and solve for bandwidth ($\boldsymbol{c}$) and the number of admitted flows ($\boldsymbol{g}$). Since the buffer B_n and QoS ϵ_n are fixed, the user n's flow has a fixed effective bandwidth $\bar{\nu}_n = \nu_n(\epsilon_n, B_n)$. We further know that $g_n \bar{\nu}_n = c_n$ at the optimal solution, i.e., the bandwidth allocated to user n will be fully utilized. Based on this, we can define a new weight coefficient $\overline{w}_n = w_n a_n / \bar{\nu}_n$, then Problem (9) can be simplified into the following form

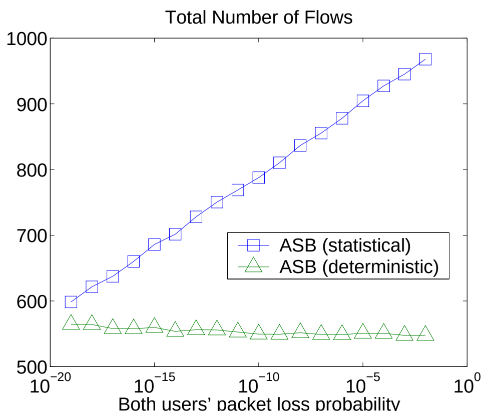
$$\begin{aligned} \text{maximize} \quad & \sum_n \overline{w}_n c_n \\ \text{subject to} \quad & \boldsymbol{c} \in \mathcal{C}, \\ \text{variables} \quad & \boldsymbol{c} \geq \mathbf{0} \end{aligned} \tag{10}$$

This becomes a standard weighted rate maximization problem, subject to the rate region constraint, and thus can be solved efficiently using the ASB algorithm in a *distributed* fashion. This determines the values of $\boldsymbol{c}$ and $\boldsymbol{g} = \boldsymbol{c}/\bar{\boldsymbol{\nu}}$. Although the ASB algorithm was originally designed to solve the spectrum management problem at the physical layer of the DSL network, it can also be used as a *mathematical machinery* to solve the statistical multiplexing Problem (10). This is made possible by using the *effective bandwidth* concept to capture the traffic characteristics and the QoS requirements.

2) Stage 2 (buffer allocation): We fix the rate allocation ($\boldsymbol{c}$), and solve for buffer allocation ($\boldsymbol{B}$) and the number of admitted flows ($\boldsymbol{g}$). The problem turns out to be a quasi-concave maximization problem, which can be solved using bisection search (by solving a sequence of feasibility problems, see [7]). In some special cases (e.g., traffic follow compound Poisson distributions with exponentially distributed file sizes), the problem is strictly concave and we can find the closed form globally optimal solution.



(a) Simulation topology



(b) Statistical multiplexing gain over deterministic services.

Fig. 6. Performance of the AM algorithm in a two-user DSL network.

D. Simulation Results

We show the performance of the AM algorithm through a simple example consisting of two ADSL modems as shown in Fig. 6(a). Here the CO line (user 1) is 5 km long and the RT line (user 2) is 3 km long. The RT is deployed 4 km downstream from the CO. We use a realistic simulator with channel gains measured from actual DSL networks.

We assume that both lines' data traffic follow the same compound Poisson distribution with exponential file size. The arrival rate λ equals 40 burst/sec and average file size $1/\mu$ equals 100 bits/burst. We consider the downstream transmission case, where a total of 20 Kbits buffer space is shared by two lines at the multiplexing link. We fix $(w_1, w_2) = (1, 1)$ and vary $\epsilon\,(= \epsilon_1 = \epsilon_2)$ from 10^{-19} to 10^{-2}. Fig. 6(b) compares the performances between statistical and deterministic services. In the deterministic service, the flows are admitted based on their peak rates[5]. By allowing a small amount of packet loss, the statistical service achieves a statistical multiplexing gain (measured in the ratio of total number of admitted flows) up to 180%.

[5]For each simulation point, we generate the compound Poisson traffic for 20000 seconds, and use the empirical peak rate as the basis for deterministic service.

VI. Space: Access Topology Design

The "space" dimension of FAST Copper consists of two types of problems: architectural decision problems, and topology design problems. The former considers with the division of functionalities between access and core networks, as discussed in Section III, e.g., how large the access networks should be, where the various types of video servers should be placed, which network elements should be responsible for mitigating crosstalk or for reducing excessive bandwidth demand, etc. Here we will focus on the more tactical problems of survivable topology design.

A. Overview

A main target of topology design in FAST project is to offer fast recovery for access networks after natural failures or malicious attacks. The access parts of the network infrastructure aggregates increasing volumes of voice, data, and video traffic from end users, but are usually the least protected, in contrast to ring-based metro networks and partial mesh backbone networks. This lack of protection makes the access network the bottleneck of end-to-end survivability, and is because of the economic reason of high price per customer when provisioning survivable networks. It is necessary to design survivable tree topologies, through the appropriate addition of a limited number of redundant links to the tree, that can provide the best survivability-cost tradeoff.

There are three types of research problems involved. First is the graph theory problem of determining a survivable tree topology. Second is the optimization problem of allocating bandwidth between primary and backup paths. Third is the systems problem of real-time signaling protocols for failure detection and recovery. We will focus only on the first problem in this paper.

Since routing capability is very expensive [1], often only the central offices can be equipped with the routing capability, and the other terminals within the access network only have very limited switching capability (such as traffic aggregation using multiplexing and demultiplexing). Accordingly, the structure of the access network is a "fat" tree rooted at central office, i.e., for an intermediate terminal node within an access network, the capacity/traffic of its upstream link is the aggregation of the capacity/traffic of all its downstream links. A sample fat tree [23] is shown at Fig. 7. Therefore, to recover from its upstream link failure, the terminal has to relay the traffic from another terminal of the same or higher level. Such feature of the access network makes the problem of designing reliable access network different from that of designing reliable backbone network.

There are four dimensions along which we can describe a taxonomy of related graph theory problems:

1) Fat-Tree Exists or Not: Network evolvability, or incremental growth of network, is an important feature in cost-effectively network provisioning. In a "brown-field" design, we often have already designed an access network without resiliency, and would like to augment it with necessary resiliency on an existing access tree to cater for user demands,

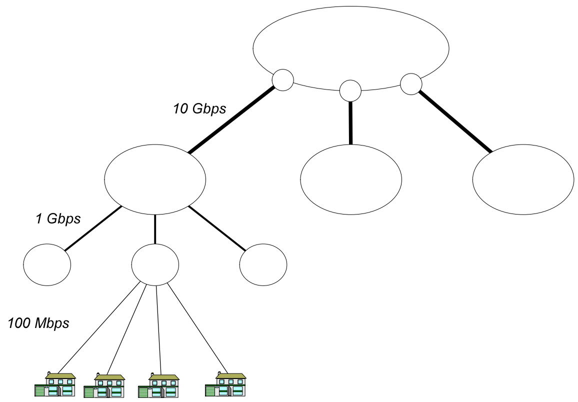


Fig. 7. Logical fat-tree architecture for access network.

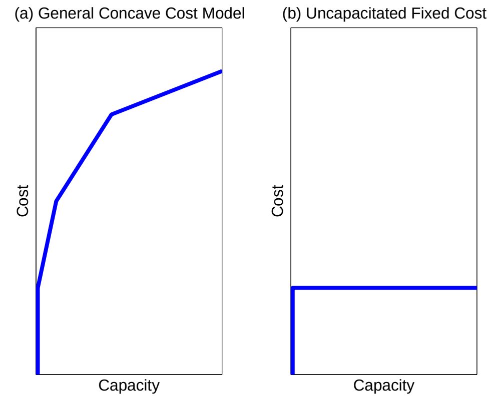


Fig. 8. Concave cost model of link capacity.

technological developments, and socio-economic trends. In a "green-field" design, it is possible to jointly design the tree and the redundant links in the first place.

2) Single Level or Multi-Level Tree: The access network between remote terminals and central office can be single level tree (e.g. star network) or multi-level tree by putting low cost and high performance switches in between.

3) Optimization (Objective-Constraint) Models: One main type of problem is to minimize the total cost to construct (and maintain) the access network by satisfying the connectivity requirement (e.g. r_i edge-disjoint paths from remote terminal i to root, and terminal i can survive against any up to $r_i - 1$ edge failures). In the other main type of problem, the network vendor can achieve a different amount of revenue by provisioning different levels of connectivity for terminal i, (i.e. r_i is a variable), the objective function could be maximizing the total revenue constrained by the limited budget in network construction.

4) Link Cost Models: Link cost is often an affine or convex function of distance, and a concave or constant function of capacity.

For example, the concave cost model (e.g. Fig. 8(a)) is mostly used in designing fiber networks, where the cost per unit length of deploying an edge is a concave non-decreasing function of the capacity/flow on the edge. The concavity is due to buy-at-bulk. The average cost per unit bandwidth of using a larger capacity cable is usually less than that of using a smaller capacity cable, e.g., one OC-12 cable is cheaper than 12 OC-1 cables. An extreme case of concave edge cost model is the uncapacitated fixed cost model (Fig. 8(b)), where the cable cost is negligible if the cost of deploying a set of cables with trenching (or hanging along poles) is dominant. With this cost model, the accurate values of traffic demands and flows on edge are not crucial since unlimited capacity is assumed.

There are 16 combinations of the above problem formulations, leading to a rich array of graph theory problems. Some of them are NP-hard, and some are under-explored new problems. We briefly discuss two of the problems in the rest of this section.

B. Budget-Constrained Revenue Maximization

In this problem, uncapacitated fixed cost model is adopted and there is no existing tree. With a limited budget in network construction, we need to search for a subgraph to maximize the total revenue from providing survivability for partial remote terminals. The problem can be proved to be NP-hard through reduction to the Steiner Tree problem [6].

For an undirected graph, $G = (V, E)$, $\hat{E}$ is the set of links where each edge consists of two unidirectional links, c_e is the cost of deploying edge e, B is the total budget, $S \subset V$ is the set of remote terminals to be connected with central office, $S - V$ consists of central office and all Steiner (optional) vertices, h_v is the revenue of providing survivability (2-connectivity) for $v \in S$, x_e is a binary variable to represent if link e is chosen in the subgraph, and r_v is a binary variable to indicate if terminal v is provided with survivability. By adopting the commodity-based formulation for Steiner network design [18], we develop an integer linear program formulation for this problem as shown in (11) below.

$$\text{maximize} \quad \sum_{v \in S} h_v r_v \tag{11a}$$

$$\text{subject to} \quad \sum_{(v,i) \in \hat{E}} f^v_{v,i} \geq r_v + 1, \tag{11b}$$

$$\sum_{(i,j) \in \hat{E}} f^v_{i,j} = \sum_{(j,k) \in \hat{E}} f^v_{j,k}, \tag{11c}$$

$$x_e \geq f^v_{i,j}, e \text{ is the undirected edge for link } (i, j), \tag{11d}$$

$$B \geq \sum_{e \in E} c_e x_e, \tag{11e}$$

$$\text{variables} \quad 1 \geq r_v \geq 0, 1 \geq f^v_{i,j} \geq 0, x_e \in \{0, 1\}. \tag{11f}$$

The above mixed integer linear programming formulation can be understood as follows. Consider the resulting subgraph with unit edge capacity and a sink at central office, assume

[6]Given a graph G with nonnegative edge costs and whose vertices are partitioned into two sets, required and Steiner (optional), find a minimum cost tree in G that contains all the required vertices and any subset of the Steiner vertices [29].

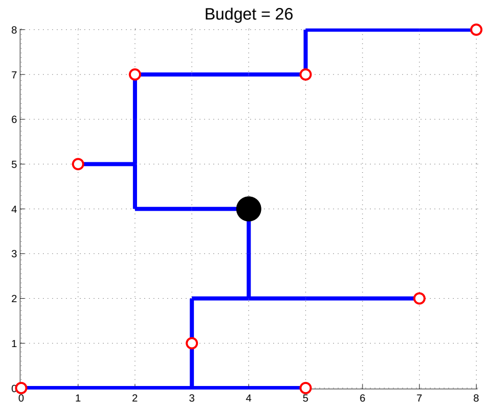


Fig. 9. Minimum budget access network design without survivability.

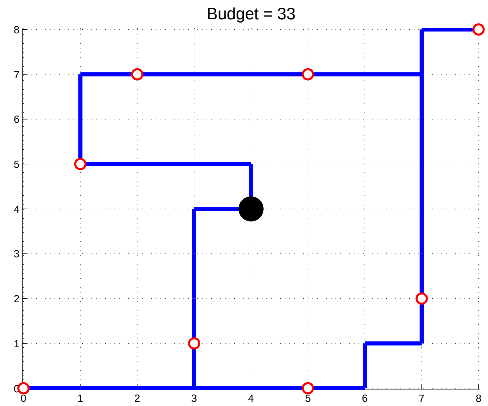


Fig. 11. Budget-constrained access network design with revenue (survivability) maximization.

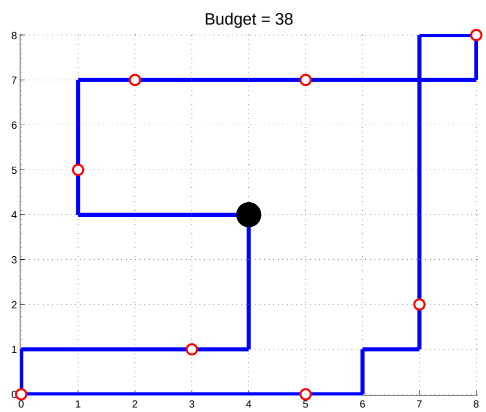


Fig. 10. Minimum budget access network design with full survivability.

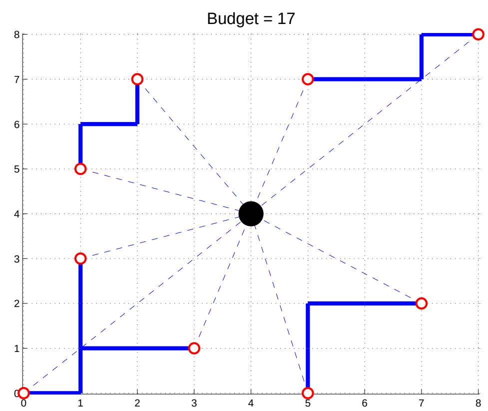


Fig. 12. Augmented topology design for an existing one-level access network to provide full survivability.

there is only one commodity initiated from a terminal $v \in S$, the max-size flow must be integral and larger than $r_v + 1$, i.e. we can relax the integral requirement for r_v. Routing for each commodity is independent of that for the other since only network connectivity is considered. Since $f_{i,j}^v$ represents the flow of commodity v on link $(i,\ j)$, (11c) demonstrates the flow conservation at an intermediate node.

By solving (11) with a commercial integer linear program solver, CPLEX 9.0, we collect some numerical results of this problem for qualitative analysis. In the sample Manhattan-like grid network shown in Figures 9, 10, and 11, the empty circles are remote terminals and the solid circle at the center is central office. Each edge has 1 unit cost and the revenue by providing survivability (2-connectivity) to each remote terminal is 1 unit (different from unit edge cost). Fig. 9 shows the minimum cost (26 units) access network design without any survivability, and Fig. 10 shows the minimum cost (38 units) access network design with full survivability (and the corresponding 8-unit revenue). In contrast, Fig. 11 shows an optimal design where the maximum revenue is 6 units under the budget constraint of 33 units.

C. Provisioning Survivability for Existing Single-Level Tree

In this problem, we search for the minimum cost incremental topology design to provide full survivability to all the remote terminals within an existing single-level access tree, which connects the central office and each remote terminal directly. Uncapacitated fixed cost model is used. To provide full survivability, we need to construct an augmented network where each terminal is connected to either one other terminal or the root. Such problem is abstracted as *Terminal Backup* problem, where we are given a graph with terminals (required vertices), Steiner (optional) vertices, and weighted edges, and the goal is to find the cheapest subgraph so that every terminal is connected to at least one other terminal for backup purpose. This problem is polynomial-time solvable [3], and a sample minimum cost augmented graph is shown as bold lines in Fig. 12, where the existing access tree is shown with dotted lines.

VII. Time: Multiuser Scheduling

A. Overview

So far we have been focusing on throughput in terms of rate region and multiplexing gain. However, there are also other

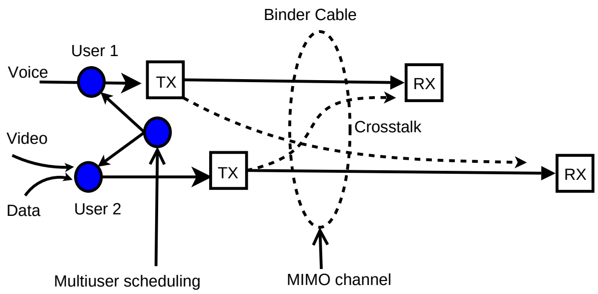


Fig. 13. Multiuser scheduling with two users in a MIMO channel with near-end crosstalk. Each user can send different number of traffic flows with different QoS requirements in the upstream and/or downstream of the binder cable.

TABLE II

DIFFERENT QOS REQUIREMENTS AND CONTROL MECHANISM. THE FIRST COLUMN SHOWS THE QOS PERFORMANCE METRIC. THE SECOND ROW SHOWS THE NATURE OF THE QOS GUARANTEE AND THE THIRD COLUMN SHOWS THE CORRESPONDING CONTROL MECHANISM IN THE DSL NETWORK.

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

QoS metrics important to triple play traffic, including delay, jitter, and fairness. These considerations in part determine which point on the rate region should the system attain, and require a finer granularity of control among the admitted flows. We also need to incorporate the stochastic dynamics of flow arrival and departure.

In this section, we discuss the issue of weight adaptation in time for throughput optimization using multiuser scheduling and other control mechanism that takes into account the QoS characteristics of traffic flows at the packet transmission level. In [6], it is shown that throughput optimization may be achieved by weighing the rate of each user transmission with suitable coefficients, which may be interpreted as reward values. Here, we will briefly outline a general problem formulation for packet scheduling in a DSL network, and a dynamic scheduling algorithm that can leverage the potential for scheduling based on the application layer characteristics and instantaneous number of flows. Figure 13 shows an overview of multiuser scheduling in a DSL network.

B. Priority Queueing of Different Traffic Flows for Each User

At each user, dynamic bandwidth or buffer allocation among flows that have different QoS requirements is necessary. Table II illustrates how different QoS requirements from each traffic flow may be met by the appropriate control mechanisms. At the inter-user level, average throughput, average delay and fairness are ensured by the multiuser scheduler. At the intra-user level, priority queueing is used to differentiate between real-time and non-real-time flows.

By properly selecting a priority function for each queue, different QoS requirements in terms of packet loss and instantaneous delay can be satisfied on a flow basis at each user. As an illustration, when a packet of a real-time flow for a particular user nears its deadline expiration, the packet receives an increasingly higher priority than other packets belonging to other flows that are in transmission. Designing appropriate priority functions for the priority queueing control mechanism based on traffic flow type and characteristics are currently under investigation.

C. Scheduling Among Users

At the inter-user level, we can design a multiuser scheduling algorithm, which maximizes the total throughput of the system and guarantees max-min fairness among the users. We first introduce the notation. Let x_n be the long-run fraction of time that the nth user transmits packets. We define the power spectral density vector $\mathbf{s}$ as $[\mathbf{s}_1^T, \ldots, \mathbf{s}_N^T]^T$ where $\mathbf{s}_k = [s_{n1}, \ldots, s_{nK}]^T$ for $n = \{1, \ldots, N\}$.

The nth user has the following feasible rate $R_n(\mathbf{s})$. The objective is the maximization of a base rate R subject to a fixed ratio between R and the long-run expected throughput of the nth user. To differentiate among user transmission requirements, we introduce a parameter β_n for the nth user. One formulation of the problem of throughput maximization through scheduling and DSM is shown as follows.

$$
\begin{array}{ll}
\text{maximize} & R \\
\text{subject to} & \beta_n R \le R_n(\mathbf{s}) x_n,\ n = 1, \ldots, N, \\
& \sum_{n=1}^{K} x_n \le 1, \\
& \sum_{k=1}^{N} s_n(k) \le P_n,\ n = 1, \ldots, N, \\
& x_n \ge 0,\ n = 1, \ldots, N, \\
& s_n(k) \ge 0,\ n = 1, \ldots, N,\ k = 1, \ldots, K, \\
\text{variables:} & R,\ x_n,\ \mathbf{s}_n, n = 1, \ldots, N.
\end{array}
\tag{12}
$$

We have designed a joint time and frequency dynamic scheduling algorithm that solves for an optimum of (12). Figure 14 shows a block diagram of the main components of the multiuser scheduling algorithm. The algorithm consists of two subproblems, namely a DSM subproblem and a flow transmission subproblem, which are solved simultaneously. Each subproblem takes as input the channel state of the MIMO channel and the flow information of all users respectively. Particularly, the DSM subproblem can be solved by the ASB algorithm. The multiuser scheduler coordinates the two subproblems through the exchange of vectors $\mathbf{w}$ and $\boldsymbol{\nu}$, and allocates a point in the Pareto boundary of the rate region to each flow of all the users. The optimal revenue vector is determined iteratively and all the flows of each user are served simultaneously. Thus, by appropriately choosing β_k for all k

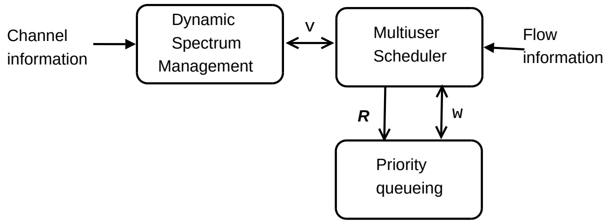


Fig. 14. A block diagram of the main components of our multiuser scheduling algorithm. The symbols $\mathbf{w}$ and $\boldsymbol{\nu}$ denote Lagrange multipliers that are exchanged between the blocks in the form of message passing. $\mathbf{R}$ denotes the base rate in (12).

at each scheduling interval, the multiuser scheduling algorithm can achieve a correspondingly wide range of revenue earned for each user.

VIII. Conclusion

FAST Copper projects presents new networking research challenges, motivates a wide range of difficult problems in fundamental research disciplines, and offers an opportunity for research to make tangible impacts on how people access information.

There are two types of research challenges in FAST Copper project. One is the array of problem formulations in the individual dimensions of Frequency, Amplitude, Space, and Time. These problems range from communication theoretic ones to graph theoretic ones, and from nonconvex optimization to stochastic systems. The other is the need to quantify architectural principles in broadband access networks. This requires a characterization of horizontal and vertical decompositions for functionality allocation. Both types of questions also have applicabilities beyond the scope of FAST Copper project itself.

In this overview paper, we have summarized the recent progress in 2005-2006 at Princeton University for the first type of challenges, and outlined the key issues involved in the second type. As FAST Copper project continues in the next few years, we expect more results and methodologies for these rich sets of research problems, and collaboration with industry towards a fast, ubiquitous, survivable, and high quality broadband access network.

Acknowledgement

We would like to thank the funding from the NSF CISE ITR program 2004-2008, collaboration with AT&T, and discussions with project co-PI John Cioff, Alexander Fraser, and their research groups. We also thank the general discussions with Elliot Anshelevich, Marc Moonen, Daniel Palomar, and Wei Yu.

References

[1] M. Andrews and L. Zhang, "The access network design problem," in *FOCS*, 1998, pp. 40–59.

[2] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar, and P. Whiting, "Scheduling in a queueing system with asynchronously varying service rates," *Probability in the Engineering and Informational Sciences*, vol. 18, pp. 191–217, 2004.

[3] E. Anshelevich, et al., "Terminal backup, 3d matching, and valid cabbage sums," preprint, 2006.

[4] T. Bonald, A. Proutiere, J. Roberts, and F. R&D, "Statistical performance guarantees for streaming flows using expedited forwarding," *IEEE INFOCOM*, vol. 2, 2001.

[5] S. Borst, "User-level performance of channel-aware scheduling algorithms in wireless data networks," in *In Proceedings of INFOCOM*, 2003.

[6] S. Borst and P. Whiting, "Dynamic channel-sensitive scheduling algorithms for wireless data throughput optimization," *IEEE Trans. on Vehicular Technology*, vol. 52, no. 3, pp. 569–586, 2003.

[7] S. Boyd and L. Vanderberghe, *Convex Optimization*. Cambridge University Press, 2004.

[8] R. Cendrillon, J. Huang, M. Chiang, and M. Moonen, "Autonomous spectrum balancing for digital subscriber lines," submitted to *IEEE Transactions on Signal Processing*, 2006.

[9] R. Cendrillon and M. Moonen, "Iterative spectrum balancing for digital subscriber lines," in *IEEE ICC*, 2005.

[10] R. Cendrillon, W. Yu, M. Moonen, J. Verlinden, and T. Bostoen, "Optimal multiuser spectrum balancing for digital subscriber lines," *IEEE Transactions on Communications*, vol. 54, no. 5, pp. 922– 933, May 2006.

[11] V. M. K. Chan, "Multiuser detection and spectrum balancing for digital subscriber lines," Master's thesis, Department of Electrical and Computer Engineering, Univeristy of Toronto, 2005.

[12] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of IEEE*, January 2007.

[13] S. T. Chung, "Transmission schemes for frequency selective gaussian interference channels," Ph.D. dissertation, Stanford University, 2003.

[14] C. Courcoubetis and R. Weber, "Buffer Overflow Asymptotics for a Buffer Handling Many Traffic Sources," *Journal of Applied Probability*, vol. 33, no. 3, pp. 886–903, 1996.

[15] J. Huang, R. Berry, and M. L. Honig, "Distributed interference compensation in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 5, pp. 1074–1084, May 2006.

[16] J. Huang, R. Cendrillon, M. Chiang, and M. Moonen, "Autonomous spectrum balancing (ASB) for frequency selective interference channels," in *IEEE ISIT*, 2006.

[17] J. Huang, C.-W. Tan, M. Chiang, and R. Cendrillon, "Statistical multiplexing over dsl networks," submitted to INFOCOM 2007.

[18] K. Jain, "A factor 2 approximation algorithm for the generalized Steiner network problem," *Combinatorica*, vol. 21, no. 1, pp. 39–60, 2001.

[19] F. Kelly, "Notes on effective bandwidths," *Stochastic Networks: Theory and Applications*, vol. 4, pp. 141–168, 1996.

[20] L. Kleinrock, "Time-shared systems: A theoretical treatment," *Journal of ACM*, vol. 14, no. 2, pp. 242–261, 1967.

[21] K. Kumaran and M. Mandjes, "Multiplexing regulated traffic streams: design and performance," *IEEE INFOCOM*, vol. 1, 2001.

[22] R. Lui and W. Yu, "Low-complexity near-optimal spectrum balancing for digital subscriber lines," *IEEE ICC*, 2005.

[23] A. Patzer, "Algorithms for a reliable access network," Master's thesis, Princeton Univeristy, 2004.

[24] M. Reisslein, K. Ross, and S. Rajagopal, "A framework for guaranteeing statistical QoS," *IEEE/ACM Transactions on Networking*, vol. 10, no. 1, pp. 27–42, 2002.

[25] S. Shakkottai, R. Srikant, and A. Stolyar, "Pathwise optimality of the exponential scheduling rule for wireless channels," *Advances in Applied Probability*, vol. 36, no. 4, pp. 1021–1045, 2004.

[26] T. Starr, M. Sorbara, J. Cioffi, and P. Silverman, *DSL Advances*. Prentice Hall, 2003.

[27] T. Starr, J. Cioffi, and P. Silverman, *Understanding Digital Subscriber Line Technology*. Prentice Hall, 1999.

[28] D. M. Topkis, *Supermodularity and Complementarity*. Princeton University Press, 1998.

[29] V. V. Vazirani, *Approximation algorithms*. New York, NY, USA: Springer-Verlag New York, Inc., 2001.

[30] D. Wischik, "Sample Path Large Deviations for Queues with Many Inputs," *The Annals of Applied Probability*, vol. 11, no. 2, pp. 379–404, 2001.

[31] W. Yu, G. Ginis, and J. Cioffi, "Distributed multiuser power control for digital subscriber lines," *IEEE Journal on Selected Areas in Communication*, vol. 20, no. 5, pp. 1105–1115, June 2002.