Autonomous Spectrum Balancing for Digital Subscriber Lines

Raphael Cendrillon, *Member, IEEE*, Jianwei Huang, *Member, IEEE*, Mung Chiang, *Member, IEEE*, and Marc Moonen, *Fellow, IEEE*

***Abstract*—The main performance bottleneck of modern digital subscriber line (DSL) networks is the crosstalk among different lines (i.e., users). By deploying dynamic spectrum management (DSM) techniques and reducing excess crosstalk among users, a network operator can dramatically increase the data rates and service reach of broadband access. However, current DSM algorithms suffer from either substantial suboptimality in typical deployment scenarios or prohibitively high complexity due to centralized computation. This paper develops, analyzes, and simulates a new suite of DSM algorithms for DSL interference-channel models called autonomous spectrum balancing (ASB). The ASB algorithms utilize the concept of a "reference line," which mimics a typical victim line in the interference channel. In ASB, each modem tries to minimize the harm it causes to the reference line under the constraint of achieving its own target data-rate. Since the reference line is based on the statistics of the entire network, rather than any specific knowledge of the binder a modem operates in, ASB can be implemented autonomously without the need for a centralized spectrum management center. ASB has a low complexity and simulations using a realistic simulator show that it achieves large performance gains over existing autonomous algorithms, coming close to the optimal rate region in some typical scenarios. Sufficient conditions for convergence of ASB are also proved.**

***Index Terms*—Digital subscriber lines (DSLs), distributed algorithm, dual decomposition, interference channel, multicarrier, power allocation, spectrum management.**

I. Introduction

A. Motivation

DIGITAL SUBSCRIBER LINE (DSL) technologies transform traditional voice-band copper channels into broadband access systems, which are typically capable of delivering data rates of several Mb/s per twisted-pair over a distance of 10 kft in the basic asymmetric DSL (ADSL). Despite over 160 million DSL lines worldwide as of 2006, the major obstacle for performance improvement in modern DSL systems remains to be *crosstalk*, which is the interference generated among different lines in the same cable binder. The crosstalk is typically 10–20 dB larger than the background noise, and direct crosstalk cancelation (e.g., [1], [2]) is difficult in many cases, due to complexity (both amount of computation needed and the requirements for new chip sets) or unbundling requirement (i.e., incumbent service providers must rent certain lines to their competitors).[1]

Recently, various dynamic spectrum management (DSM)[2] algorithms have been proposed to address this frequency-selective interference problem by dynamically optimizing transmission power spectra of different modems in DSL networks. DSM algorithms can significantly improve data rates over the current practice of static spectrum management, which mandates spectrum mask or flat power backoff across all frequencies (i.e., tones).

This paper develops, analyzes, and simulates a suite of DSM algorithms for power allocation (or, equivalently, bit loading), called autonomous spectrum balancing (ASB). Overcoming the bottlenecks in the state-of-the-art DSM algorithms, ASB is a set of algorithms that, simultaneously, is autonomous (distributed algorithm across the users without explicit real-time information exchange), has low complexity, is provably convergent under certain sufficient conditions, and achieves rate region close to the global optimum. The methods of "static pricing" and "frequency-selective waterfilling" developed in ASB may also be of interest to the general problems of decoupling coupled objective function and of multicarrier interference channel.

B. Related Work on DSM Algorithms

One of the first DSM algorithms is the *Iterative Water-filling* (IW) algorithm [3], where each line maximizes its own data rate by waterfilling over the noise and interference from other lines. The IW algorithm is autonomous, has a linear complexity in the number of users and number of frequency tones, and has been shown to converge in typical DSL deployments, e.g., [3], [4]. Although IW can achieve near optimal performance in weak interference channels, it is highly-suboptimal in the widely-encountered near–far scenarios (which will be described in detail in Section II), such as mixed central office and remote terminal

Manuscript received September 1, 2006; revised November 30, 2006. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Brian L. Evans. This work was supported in part by Alcatel-Bell and by the US NSF Grant CNS-0427677 and CCF-0448012. A portion of this paper was has appeared in the *IEEE Proceedings of the International Symposium of Information Theory*, July 2006.

R. Cendrillon is with Marvell Hong Kong, Ltd., Mongkok, Hong Kong (e-mail: raphael@cendrillon.org; raphaelc@marvell.com).

J. Huang and M. Chiang are with Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: jianweih@princeton.edu; chiangm@princeton.edu).

M. Moonen is with the Department of Electrical Engineering, Katholieke Universiteit Leuven (K.U. Leuven,), ESAT/SISTA, B-3001 Leuven-Heverlee, Belgium (e-mail: moonen@esat.kuleuven.be; Marc.Moonen@esat.kuleuven.be).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2007.895989

[1]Although in an unbundled network DSM can be applied to in-domain lines, in many cases out-of-domain lines cannot be coordinated, leading to some suboptimality. Similarly the network management center can be used to coordinate lines in a centralized fashion, however such a network management center would require full knowledge of the network topology, which is often difficult to implement in practice. Further discussion can be found in Section I-B.

[2]The DSM algorithms discussed in this paper are different from the "dynamic spectrum sharing" algorithms, which are used to refer to opportunistic sharing of the spectrum resources in wireless communications.

TABLE I
COMPARISON OF VARIOUS DSM ALGORITHMS

Algorithm	Operation	Complexity	Performance	Reference
Synchronous Case				
IW	Autonomous	$O(KN)$	Suboptimal	[3]
OSB	Centralized	$O\left(Ke^N\right)$	Optimal	[5]
ISB	Centralized	$O\left(KN^2\right)$	Near optimal	[6], [7]
ASB-S1	Autonomous	$O(KN)$	Near optimal	this paper
Asynchronous case				
Greedy algorithm	Centralized	$O\left(N^2K^3\right)$	Suboptimal	[10]
ASB-A1	Autonomous	$O\left(NK^2\log_2(K)\right)$	Suboptimal	this paper

deployments of ADSL and upstream VDSL. This is in part due to the greedy and selfish nature of the algorithm.

Recently two optimal but centralized DSM algorithms were proposed, the *Optimal Spectrum Balancing* (OSB) algorithm [5] and the *Iterative Spectrum Balancing* (ISB) algorithm [6], [7]. The OSB algorithm addresses the spectrum management problem through the 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. OSB has an exponential complexity in the number of users, making it intractable for DSL network with more than five lines. As an improvement over the OSB algorithm, ISB was proposed to implement the weighted-rate sum optimization in an iterative fashion over the users. This leads to a quadratic complexity in the number of users, which makes the ISB feasible for networks with a relatively large number of users.

However, an even more critical issue is that both OSB and ISB are centralized algorithms, which rely on a centralized network management center (NMC) to optimize the power spectral density (PSD) for all modems. The NMC requires knowledge of the crosstalk channels among all lines and all background noise. Identification and transmission of crosstalk channel measurements back to the NMC are not supported in existing standards either. The operation of NMC requires a lot of overhead, in terms of both bandwidth and infrastructure. Furthermore, regulatory requirements on unbundling service make it impossible to perform a centralized optimization. Finally, many lines in the same binder terminate on different quad cards in the DSL Access Multiplexer because customers in the same neighborhood sign up for service at different times, which makes it hard to have central coordination even if one can tolerate the costs.

A semi-centralized DSM algorithm called SCALE is proposed in [8]. SCALE algorithm achieves better performance than IW with comparable complexity. However, the algorithm is not autonomous since explicit message passing among users is required. Such explicit real-time message passing in an uncoordinated fashion requires modems to have sophisticated processing capabilities not available in DSL modems, including blind synchronization, blind identification of the crosstalk channel, blind detection of the transmit constellation used by the crosstalk, and blind detection of the crosstalk signal.

The band preference method is a practical way of implementing an optimized DSM PSD in a distributed fashion [9]. While the band preference method calculates the bitloading in a distributed fashion, the band-preference coefficients (which correspond to a spectral mask imposed on the waterfilling algorithm) need to be calculated in some way, centralized or distributed. This often requires the use of a centralized spectrum management center. The performance of the band preference method depends on the choice of the specific spectrum management algorithm used.

IW, OSB, ISB, and SCALE mentioned above all assume synchronous transmissions of the modems, which allows crosstalk to be modeled independently on each tone. This synchronization is rarely true in practice. Instead, 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) complicates the DSM problem further. The state-of-the-art results for asynchronous transmissions are the two centralized greedy algorithms proposed in [10], bit-subtracting and bit-adding algorithms. Both algorithms start from the power spectral density (PSD) obtained with the ISB algorithm in the synchronous case, and search for local optimal solutions in the neighborhood by taking ICI into account. But again these are centralized algorithms.

C. Summary of Contributions

The advantages of ASB algorithms are summarized as follows. First of all, ASB is autonomous: it can be applied in a distributed fashion across users with no explicitly information exchange in real-time. Furthermore, the algorithm has low complexity in both the number of users and tones, and is proved to be convergent under certain sufficient conditions on the channel gains. In the synchronous case, the ASB algorithm has a similar complexity as IW, but in the near–far scenario achieves a performance much better than IW and very close to ISB and OSB. In the asynchronous case, the ASB algorithm reduces the complexity from those in [10], and achieves significant better performance than the ASB algorithm that does not consider the ICI. These features are obtained despite the convexity and coupling in the optimization problem of DSM. The comparisons between ASB algorithms and other existing algorithms are listed in Table I. It compares various aspects of different DSM algorithms, where ASB attains the best tradeoff among distributiveness, complexity, and performance. Here we use K to denote the number of tones and N to denote the number of users.

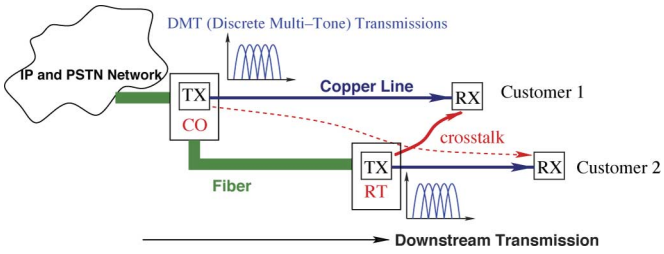


Fig. 1. Mixed CO/RT distribution.

The key idea behind ASB is to leverage the fact that DSL interference channel gains are very slowly time-varying, which enables an effective use of the concept of "reference line" that represents a typical victim line. Roughly speaking, the reference line represents the statistical average of all victims within a typical network, which can be thought as a "static pricing". This differentiates the ASB algorithm with power control algorithms in the wireless setting, where pricing mechanisms have to be adaptive to the change of channel fading states and network topology, or Internet congestion control, where time-varying congestion pricing signals are used to align selfish interests for social welfare maximization. By using static pricing, no explicit message passing among the users is needed and the algorithm becomes autonomous across the *users*. When adapting its PSD, each line attempts to achieve its own target rate while minimizing the damage it does to the reference line. We show such mechanisms can attain the balance between selfish and socially responsible operation. At the same time, each user in ASB still keeps a local "dynamic pricing" of the individual power constraint, which enables its own optimization problem to be decoupled across the *tones* within each user. We prove the convergence of ASB under an arbitrary number of users, for both sequential and parallel updates. Since IW can be recovered as a special case of ASB in the synchronous case, our proof techniques extend those in previous work on IW [3], [11].

The rest of the paper is organized as follows. We introduce the system model in Section II, for both synchronous and asynchronous transmission cases. The spectrum management problem and a general framework of ASB are outlined in Section III. ASB algorithms for the synchronous and asynchronous cases will be given in Sections IV and V, respectively. We provide convergence proofs and simulation results in Sections VI and VII. The complexity properties of the ASB algorithm and the IW algorithm are given in Section VIII, and we conclude in Section IX.

II. System Model

ASB can be applied to many network topologies. In this paper we will often examine a typical near–far deployment for downstream ADSL transmissions with a frequency band up to 1.1 MHz,[3] as shown in Fig. 1, since it is one of the scenarios where DSM techniques can give significant performance improvement. In this scenario there are at least 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 crosstalk (interference) to each other on all frequency tones. Although the 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 and since the two lines are in the same binder. However, CO TX generates little crosstalk to RT RX due to the long distance between them.

A similar near–far problem also occurs in the upstream transmission for VDSL, which operates at a frequency band up to 12 MHz, and line lengths are typically limited to less than 1.2 km [12], [13]. As a result, VDSL modems are typically deployed at one point in the network (e.g., a RT node), thus do not have the mixed CO/RT problem in the downstream transmissions. However, due to the difference in customer home locations, shorter lines exhibit strong crosstalks into the longer lines receivers in the upstream transmissions. Furthermore, in mixed VDSL/ADSL deployments, RT-deployed VDSL will interfere with the CO-deployed ADSL signals in the downstream.

Next we introduce the mathematical models for both the synchronous and asynchronous transmission cases, following the notation in [5], [6], and [10].

A. Synchronous Transmission

Consider a DSL network with a set of $\mathcal{N} = \{1, \ldots, N\}$ users (i.e., lines, transmitting modems) and $\mathcal{K} = \{1, \ldots, K\}$ tones (i.e., frequency carriers). Assuming the standard *synchronous* discrete multi-tone (DMT) modulation, transmissions can be modeled independently on each tone k as follows:

$$\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{z}_k.$$

The vector $\boldsymbol{x}_k \triangleq \{x_k^n, n \in \mathcal{N}\}$ contains transmitted signals on tone k, where x_k^n 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; $\boldsymbol{z}_k$ 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 $h_k^{n,m}$. We denote the transmit Power Spectral Density (PSD) $s_k^n \triangleq \mathcal{E}\{|x_k^n|^2\}/\Delta_f$, where $\mathcal{E}\{\cdot\}$ denotes expected value, and Δ_f denotes inter-carrier spacing. The vector containing the PSD of user n on all tones as $\boldsymbol{s}^n \triangleq \{s_k^n, k \in \mathcal{K}\}$.

When the number of interfering users is large, the interference can be well approximated by a Gaussian distributed random variable. The achievable bit rate of user n on tone k is defined as

$$b_k^n \triangleq \log\left(1 + \frac{1}{\Gamma}\frac{s_k^n}{\sum_{m \neq n} \alpha_k^{n,m} s_k^m + \sigma_k^n}\right) \quad (1)$$

[3]The near–far problem does not occur in the upstream ADSL case, where the transmission frequency band is below 138 kHz and crosstalk is minimal at such low frequencies.

where $\alpha_k^{n,m} \triangleq |h_k^{n,m}|^2/|h_k^{n,n}|^2$ is the normalized crosstalk channel gain from user m to user n, and $\sigma_k^n \triangleq \mathcal{E}\{|z_k^n|^2\}/|h_k^{n,n}|^2\Delta_f$ is the normalized noise power density. Here Γ denotes the SINR-gap to capacity, which is a function of the desired BER, coding gain and noise margin [14]. For notational simplicity, we absorb Γ into the definition of $\alpha_k^{n,m}$ and σ_k^n. 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: $\sum_{k\in\mathcal{K}} s_k^n \le P^n$. The data rate on line n is thus $R^n = \sum_{k\in\mathcal{K}} b_k^n$.

B. Asynchronous Transmission

In practice, it is often difficult to maintain perfect synchronization between different DMT blocks due to different transmission delays on different lines. Compared with the synchronous transmission case, here the received PSD of user n on tone k, $\mathcal{E}\{|y_k^n|^2\}$, also depends on other users' transmit PSD on tones *other than* tone k,

$$\mathcal{E}\left\{|y_k^n|^2\right\} = |h_k^{n,n}|^2 s_k^n + \sum_{m\neq n}\left(\sum_{j=1}^{K}\gamma(k-j)\left|h_j^{n,m}\right|^2 s_j^m\right) + \mathcal{E}\left\{|n_k^n|^2\right\}.$$

Here $\gamma(j)$ is the ICI coefficients estimated in the worst case [10],

$$\gamma(j) = \begin{cases} 1, & j = 0 \\ \frac{2}{K^2\sin^2\left(\frac{\pi}{K}j\right)}, & -\frac{K}{2} \le j < \frac{K}{2}, j \neq 0 \end{cases}$$

and has the symmetric and circular properties, i.e., $\gamma(-j) = \gamma(j) = \gamma(K-j)$. Then the achievable bit rate of user n on tone k in (1) needs to be revised as (with Γ set to 1)

$$b_k^n \triangleq \log\left(1 + \frac{s_k^n}{\sum_{m\neq n}\left(\sum_{j=1}^{K}\gamma(k-j)\alpha_{k,j}^{n,m}s_j^m\right) + \sigma_k^n}\right) \tag{2}$$

where $\alpha_{k,j}^{n,m} \triangleq |h_j^{n,m}|^2/|h_k^{n,n}|^2$. All the other system parameters and constraints are the same as the synchronous case.[4]

III. Spectrum Management Problem and the General Framework of ASB

We consider the following spectrum management problem

$$\max_{\{\boldsymbol{s}^n, n\in\mathcal{N}\}} R^1 \quad \text{s.t. } R^n \ge R^{n,\text{target}}, \forall n > 1 \quad \text{s.t. } \sum_{k\in\mathcal{K}} s_k^n \le P^n, \forall n. \tag{3}$$

[4]While windowing [15] at the transmitter and receiver can be used to lower the DMT sidelobes and help reject ICI, in our experience a high level of ICI still remains, leading to significant performance degradation. Thus it is an important problem to mitigate ICI through DSM techniques.

Here $R^{n,\text{target}}$ denotes the target rate of user n, and we can pick an arbitrary user to be user 1. Due to interference between users, Problem (3) is nonconvex. Furthermore, it is highly coupled across users (due to crosstalk) and tones (due to total power constraint as well as ICI in the asynchronous case), making it a very difficult optimization problem to solve.

The rate region achieved by all users is convex in the asymptotic case when number of tones becomes large [5]. Thus by changing the values of $R^{n,\text{target}}$ of all users $n > 1$, the solutions of Problem (3) can trace out the Pareto optimal boundary of the rate region.

It appears that any algorithm that globally solves (3) 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 (3), each user adopts a PSD that achieves a fair compromise between maximizing their own data-rate and minimizing the damage they do to other users. Based on this insight, we introduce the concept of reference line, a virtual line that represents a typical victim user within the DSL system. One choice, but not the only one, for the reference line is to set it as the longest line seen within a network, which tends to have the weakest direct channel and see the worst crosstalk spectrum. Then, instead of solving (3), each user tries to maximize the achievable rate on the reference line, subject to its own rate and total power constraints.

Note that the reference line is a fictitious line, and is used to represent a typical victim in a DSL network. This is independent of a particular binder, as no specific knowledge of a binder's configuration is assumed. As such no centralized control is necessary, and the algorithm can be implemented in an autonomous fashion. The only knowledge a modem needs is its direct channel, background noise and the distance from the CO to the RT if it is RT distributed. All of this information can either be measured locally, or, in the case of the CO to RT distance, can be programmed at the time that the RT is installed. This allows ASB to be implemented in an autonomous fashion during run-time, with the PSD and bitloading calculated locally.

Since the main purpose of introducing the reference line is to characterize the damage that each user does to other interfering users, 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 $R^{n,\text{ref}} \triangleq \sum_{k\in\mathcal{K}} \tilde{b}_k^n$, where the achievable bit rate on tone k in the synchronous case is defined as

$$\tilde{b}_k^n \triangleq \log\left(1 + \frac{\tilde{s}_k}{\tilde{\alpha}_k^n s_k^n + \tilde{\sigma}_k}\right) \tag{4}$$

and, in the asynchronous case, as

$$\tilde{b}_k^n \triangleq \log\left(1 + \frac{\tilde{s}_k}{\sum_{j=1}^{K}\gamma(k-j)\tilde{\alpha}_{k,j}^n s_j^n + \tilde{\sigma}_k}\right). \tag{5}$$

Intuitively, the reference line serves as a penalty term in each user's optimization problem to align selfish behavior with social

welfare maximization, and eliminates the need of explicit message passing among users. Thus, instead of solving Problem (3) which requires global information, we let each user n solve the following problem in ASB algorithm:

$$\max_{\boldsymbol{s}^n} R^{n,\mathrm{ref}} \quad \text{s.t. } R^n \geq R^{n,\mathrm{target}} \quad \text{s.t. } \sum_{k\in\mathcal{K}} s_k^n \leq P^n. \qquad \text{(OPT1)}$$

We want to emphasize that the each user autonomously solves a different version of Problem (OPT1). For user n, Problem (OPT1) only involves optimization over its own PSD $\boldsymbol{s}^n$, which determines the achieved rates of itself (R^n) and the reference line $(R^{n,\mathrm{ref}})$. The interference generated by other users are considered as fixed background noise in the optimization, and the achieved rates of other users in the network do not need to be considered. After each user solves its own version of Problem (OPT1), the crosstalk values change accordingly. Then each user n has to solve Problem (OPT1) again, repeating the process until the PSD converges. The complete ASB algorithms will be given the Sections IV and V, where each version of ASB deploys a unique way of solving Problem (OPT1). In Section VII, we will use "area of the rate region" as the performance metric when comparing ASB algorithms with other existing DSM algorithms (e.g., [3], [5]–[7], [10]).

To facilitate the analysis in the following sections, we also consider a variation of Problem (OPT1), where we relax user n's target rate constraint and replace the optimization objective by a weighted rate sum of user n's own rate and the reference line's rate seen by user n, i.e.,

$$\max_{\boldsymbol{s}^n} w^n R^n + (1-w^n) R^{n,\mathrm{ref}} \text{ s.t. } \sum_{k\in\mathcal{K}} s_k^n \leq P^n. \quad \text{(OPT2)}$$

Here the weight parameter $w^n \in [0,1]$, where $w^n = 1$ means user n performs a pure selfish optimization, and $w^n = 0$ means the reference line's rate will be maximized.[5] In the synchronous case, it has been shown in [5] that the rate region of Problem (OPT1) (in terms of R^n and $R^{n,\mathrm{ref}}$) is convex in the asymptotic case with large number of tones. We can always find a value of w^n such that the optimal result of Problem (OPT2) is the same as that of Problem (OPT1) (i.e., find a w^n such that the solution of Problem (OPT2) satisfies $R^n = R^{n,\mathrm{target}}$) as long as the latter is feasible. Thus the key challenge of the ASB algorithm is to efficiently solve Problem (OPT2). The above correspondence is not necessarily true in the asynchronous case. In that case, we can still use Problem (OPT2) as an approximation of Problem (OPT1) to derive an algorithm that achieves good performance.

Remark 1: The crosstalk channels into the reference line $\tilde{\alpha}_k^n$ and $\tilde{\alpha}_{k,j}^n$ are modeled using the empirical models that have been developed within the standards [12], [13], [16]. These are based on extensive field measurements and give a good representation of the typical crosstalk channels seen in practice. Alternatively, it is also possible for the operator to use their own crosstalk channel models based on measurements made within their specific network, or with more advanced channel models which take into account both the inter-pair distance and non-ideal twisting of the twisted pairs within a binder [17]. For the empirical models used in standards, the only information needed to calculate the crosstalk channel is the length of the reference line, and the distance from the CO to the RT if a modem is RT distributed. All this information can be pre-set by the network operator at the time that a modem is installed. Although it may be possible to update this information periodically over the timescales of months or longer, such procedures are not required for the operation of the ASB algorithm.

Remark 2: The ASB algorithms use a static background noise spectrum for the reference line $\widetilde{\sigma}_k$, which is set to the line noise seen by the reference line in the absence of self-crosstalk, i.e., crosstalk from other DSL systems. In our experience, using this choice for the reference noise leads to good performance in a broad range of scenarios. We believe the reason for this is that in most typical DSL deployments, the shorter lines, which could potentially cause severe crosstalk to the weaker lines in the system, will be configured such that they reduce their PSDs in the frequencies where the weaker lines are active. As a result, in a DSL deployment with a reasonable distribution of rates, each line should expect to see only a marginal increase in its background noise spectrum due to crosstalk from the other lines in the system. This provides an intuitive explanation why the choice of self-crosstalk-free reference noise yields good performance. Mathematically, it means that the specific engineering problem structures in this non-convex and coupled optimization problem can be leveraged to provide a very effective approximation solution algorithm.

Remark 3: The reference line transmit PSD $\tilde{s}_k$ is also static, and is set to the PSD adopted by the reference line in the absence of self-crosstalk, and with a background noise of $\widetilde{\sigma}_k$. This PSD will be set based on the spectrum adaptation algorithm running on the modem when it operates in a fully selfish mode. In the simulations later in this paper, we use conventional single-user waterfilling to set $\tilde{s}_k$, although in principle any static spectrum management algorithm could be used. Although this choice of reference noise and reference line transmit PSD is suboptimal, it allows for an autonomous implementation, and as shown in Section VII, leads to a significant performance improvement over state-of-the-art autonomous algorithms, and in some scenarios leads to near-optimal performance.

IV. ASB Algorithms in Synchronous Transmission

In this section, we develop an ASB algorithm for the synchronous case, where the achievable bit rates of user n and the reference line (from user n's perspective) are given by (1) and (4), respectively. Since the transmissions on different tones are orthogonal to each other here, we can use dual decomposition [18] to solve Problem (OPT2), defined for each user n. Although Problem (OPT2) is nonconvex, we know from [5] that the corresponding duality gap of Problem (OPT2) is zero in the asymptotic case where the total number of tones is large, thus

[5]Problem (OPT2) can be derived from Problem (OPT1) using standard Lagrangian relaxation of user n's target rate constraint, where the dual variable is chosen to be $w^n/(1-w^n)$, which ranges from 0 to ∞. This weighted rate maximization representation was also used in [6] and [8].

solving the dual problem can lead to optimal primal solution. We name the algorithm in this section as ASB-S1, where we solve Problem (OPT2) through a dual decomposition. Each user n solves Problem (OPT2) by solving a nonconvex problem on each of the K tones and choosing the dual variable (i.e., dynamic price) such that the total power constraint is tight. Then users take turns to perform this optimization until the PSDs converge.

By incorporating the total power constraint into the objective function, we have the following relaxation of Problem (OPT2):

$$L^n \triangleq (1-\lambda^n)\left(w^n R^n + (1-w^n)R^{n,\mathrm{ref}}\right) - \lambda^n \sum_{k\in\mathcal{K}} s_k^n.$$

Here $\lambda^n \in [0,1]$ and needs to be chosen such that $(\sum_k s_k^n - P^n)\lambda^n = 0$. Then Problem (OPT2) can be solved by the following unconstrained optimization problem:

$$\max_{\boldsymbol{s}^n} L^n(w^n, \lambda^n, \boldsymbol{s}^n, \boldsymbol{s}^{-n}) \tag{6}$$

where $\boldsymbol{s}^{-n} = (\boldsymbol{s}_k^1, \ldots, \boldsymbol{s}_k^{n-1}, \boldsymbol{s}_k^{n+1}, \ldots, \boldsymbol{s}_k^N)$ denotes the PSD of all users except user n. Further define

$$L_k^n = (1-\lambda^n)\left(w^n b_k^n + (1-w^n)\tilde{b}_k^n\right) - \lambda^n s_k^n \tag{7}$$

then it is clear that L^n can be decomposed into a sum across tones of L_k^n, $L^n = \sum_k L_k^n$. As a result, Problem (6) can be decomposed into K subproblems, one for each tone k. The optimal PSD that maximizes L_k^n is

$$s_k^{n,S1} = \arg \max_{s_k^n \in [0,P^n]} L_k^n\left(w^n, \lambda^n, s_k^n, s_k^{-n}\right) \tag{8}$$

where $s_k^{-n} = (s_k^1, \ldots, s_k^{n-1}, s_k^{n+1}, \ldots, s_k^N)$. Although L_k^n is nonconvex in s_k^n, the maximization is over a scalar variable only, and the optimal value $s_k^{n,S1}$ can be easily found as follows. First solve the first order condition, $\partial L_k^n / \partial s_k^n = 0$, which is equivalent to

$$\frac{(1-\lambda^n)w^n}{s_k^{n,S1} + \sum_{m\neq n}\alpha_k^{n,m} s_k^m + \sigma_k^n} - \frac{(1-\lambda^n)(1-w^n)\tilde{\alpha}_k^n \tilde{s}_k}{\left(\tilde{s}_k + \tilde{\alpha}_k^n s_k^{n,S1} + \tilde{\sigma}_k\right)\left(\tilde{\alpha}_k^n s_k^{n,S1} + \tilde{\sigma}_k\right)} - \lambda^n = 0. \tag{9}$$

Equation (9) can be simplified into a cubic equation which has three roots that can be written in close form. Then comparing the value of L_k^n at each of these three roots, as well as checking the boundary solutions $s_k^n = 0$ and $s_k^n = P^n$, we can find out the corresponding value of $s_k^{n,S1}$.[6]

[6]If an integer bitloading constraint is applied, then we can simply calculate the PSD required to support each integer bitloading, and then evaluate the objective function L_k^n at the PSD corresponding to each integer bitloading value. The optimal choice is then selected. This allows integer bitloading constraints to be incorporated without increasing complexity. Furthermore, spectral mask constraints can also be applied in a straightforward fashion by disregarding any solution to the cubic equation that lies above the spectral mask, and adding the spectral mask level itself to the set of points evaluated in the optimization.

User n then updates λ^n to enforce the total power constraint, and updates w^n to enforce the target rate constraint. Both parameters can be found by a simple bisection search. Users then iterate until all PSDs converge. The complete ASB-S1 algorithm is given in Algorithm I.

Algorithm 1: ASB Synchronous Model Version 1 (ASB-S1)

Initialize PSDs: $\boldsymbol{s}_k^n \leftarrow P^n/K, \forall n \in \mathcal{N}, k \in \mathcal{K}$.
repeat
  for all user $n \in \mathcal{N}$ **do**
    Initialize $w_{\min}^n = 0$, $w_{\max}^n = 1$
    while $|\sum_k b_k^n - R^{n,\mathrm{target}}| > \epsilon$ **do**
      $w^n = (w_{\min}^n + w_{\max}^n)/2$
      Initialize $\lambda_{\min}^n = 0$, $\lambda_{\max}^n = 1$
      while $|\sum_k s_k^n - P^n| > \epsilon$ **do**
        $\lambda^n = (\lambda_{\min}^n + \lambda_{\max}^n)/2$
        $s_k^n \leftarrow \arg\max_{s_k^{n\prime} \in [0,P^n]} L_k^n, \forall k \in \mathcal{K}$.
        if $\sum_k s_k^n > P^n$ **then**
          $\lambda_{\min}^n = \lambda^n$
        else
          $\lambda_{\max}^n = \lambda^n$
        end if
      end while
      if $\sum_k b_k^n > R^{n,\mathrm{target}}$ **then**
        $w_{\max}^n = w^n$
      else
        $w_{\min}^n = w^n$
      end if
    end while
  end for
until all users' PSDs converge

Remark 4: The ASB algorithm leverages strong design points from both OSB and IW. Like OSB, ASB uses a weighted rate-sum to account for the damage done to other lines within the network when optimizing each line's spectra. This weighted rate-sum leads to a significant performance improvement over IW and in some scenarios leads to near-optimal performance. Like IW, ASB uses an iterative approach, optimizing the PSD of each user in turn.

Remark 5: The concept of a reference line has been employed extensively in heuristic-based DSM algorithms in the industry, including the reference PSD method that is currently mandated in the VDSL standards [12], [13], [16]. The reference PSD method is used in upstream VDSL transmissions to mitigate the near–far problem. A similar technique has also been recommended for downstream transmissions in order to protect existing ADSL services from RT distributed VDSL [19]. There is a strong connection between ASB and the reference PSD heuristics adopted in standards. Although the technique for optimizing the PSD in ASB is different to that in the reference PSD method, both algorithms use representative "reference line" that shows what a typical line in the network looks like, and how it should be expected to behave. In this paper, we also develop proofs for convergence properties of ASB.

Remark 6: In considering only a single reference line, the ASB algorithm makes an implicit assumption that, by protecting the typical victim line in the binder, a user will indirectly protect other shorter lines (i.e., stronger lines). The ASB algorithm could be extended in a straightforward way to include multiple reference lines, which does not impact the convergence properties and only leads to a small increase in complexity. For each extra reference line introduced into ASB, an extra local maxima will appear in the optimization of (8). ASB algorithm evaluates the objective function at each local maxima and chooses the global maximum. As the frequency increases, we observe that the global optimal solution chosen by the ASB algorithm jumps from a lower local optimal solution to a higher one. This is because, as frequency increases, the longest reference lines becomes inactive due to weak direct channel in the high frequency band, thus it is no longer necessary to protect this line. A higher PSD is then chosen that corresponds to a higher local optima. This new PSD will protect the second longest reference line, which is now the weakest line in the system for that particular frequency. When there are M reference lines, the ASB objective function exhibits up to $M+1$ local maxima. The first M local maxima correspond to protecting each of the reference lines, while the $(M+1)$st local maxima corresponds to the completely selfish waterfilling solution, which is employed in the very highest frequencies when all reference lines have switched off due to weak direct channels. To simplify presentation, in this paper we only focus on the approach of using a single reference line.

V. ASB Algorithms in Asynchronous Transmission

In this section, we propose an ASB algorithm for the asynchronous case, where the achievable bit rates of user n and the reference line (from user n's perspective) are given by (2) and (5). In this case, Problem (OPT2) is still non-convex and highly coupled due to crosstalk. Different from the synchronous case, a dual-based decomposition is not even applicable here since the PSD across different tones are coupled due to ICI.

We will introduce a greedy power shuffle algorithm into the ASB framework, where each user n first initializes the PSD level by solving Problem (OPT2) assuming synchronous transmission (i.e., temporarily ignoring the ICI), then shuffle its PSD $\boldsymbol{s}^n$ (i.e., subtract a small amount from one tone and add it back to another tone) to reach a locally optimal solution of Problem (OPT2). Each user takes turns to perform this power shuffling until the PSDs converge.

Let's denote the objective function of Problem (OPT2) as

$$J^n(\boldsymbol{s}^n) = w^n \sum_k b_k^n\left(s_n^k\right) + (1-w^n)\sum_k \tilde{b}_k^n(\boldsymbol{s}^n).$$

For notational simplicity, in the above expression we ignore the dependence of J^n on $\boldsymbol{s}^{-n}$ (which is assumed to be fixed during user n's PSD optimization). Define $\triangle s$ as the incremental amount of power a user can change on a tone at a time. In other words, $\triangle s$ represents the granularity of the power shuffle, which trades off performance and convergence speed.

For each user n with fixed w^n, each search iteration consists of two phases: *subtraction phase* and *addition phase*. In the subtraction phase, user n reduces its PSD by $\triangle s$ on the tone that yields the maximum increase in $J^n(\boldsymbol{s}^n)$ (or the smallest decrease if decreasing $\triangle s$ on any tone leads to a decreased objective). In the addition phase, user n increases its PSD by $\triangle s$ on the tone that yields the maximum increase in $J^n(\boldsymbol{s}^n)$ (or smallest decrease similar as in the subtraction phase). This iteration repeats until the *net change* of $J^n(\boldsymbol{s}^n)$ in the last iteration (i.e., the sum of changes in both phases) is zero. Note that the net change of objective function will never be negative in a single iteration, since in the addition phase a user can always add $\triangle s$ back to the same tone chosen in the subtraction phase and recover the PSD level as in the previous iteration.

The complete ASB-A1 algorithm is given in Algorithm 2. Line 7 computes user n's PSD similar as in the synchronous case, given fixed transmission PSDs of other users, $\boldsymbol{s}^{-n}$. Lines 8 to 10 refine the value of $\boldsymbol{s}^n$ several times by taking ICI into explicit consideration. For each value of granularity $\triangle s$, we apply the power shuffle (PS) subroutine (Algorithm 3) to update $\boldsymbol{s}^n$ until convergence is reached, which occurs once no further greedy power swap can increase the objective. In a similar fashion to the barrier method [20], we use the optimal solution from the previous refinement as the initial position in the current refinement. By using diminishing values of $\triangle s$, we achieve a much faster convergence rate and higher accuracy than can be achieved with a single PSD granularity. Finally, user n updates w^n in lines 11 to 15 using bisection search to make the target rate constraint tight.

Algorithm 2: ASB Asynchronous Model Version 1 (ASB-A1)

1: Initialize PSDs: $\boldsymbol{s}_k^n \leftarrow P^n/K, \forall n \in \mathcal{N}, k \in \mathcal{K}$.
2: **repeat**
3: **for all** user $n \in \mathcal{N}$ **do**
4: Initialize $w_{\min}^n = 0, w_{\max}^n = 1$
5: **while** $|\sum_k b_k^n - R^{n,\text{target}}| > \epsilon$ **do**
6: $w^n = (w_{\min}^n + w_{\max}^n)/2$
7: Compute $\boldsymbol{s}^n$ as Lines 7 to 16 in ASB-S1
8: **for all** $\triangle s = 0, -1, \ldots, -100\,\text{dBm/Hz}$ **do**
9: $\boldsymbol{s}^n \leftarrow PS(n, w^n, \boldsymbol{s}^n, \boldsymbol{s}^{-n}, \triangle s)$.
10: **end for**
11: **if** $\sum_k b_k^n > R^{n,\text{target}}$ **then**
12: $w_{\max}^n = w^n$
13: **else**
14: $w_{\min}^n = w^n$
15: **end if**
16: **end while**
17: **end for**
18: **until** all users' PSDs converge

Algorithm 3: Power Shuffle (PS) subroutine

1: **procedure** $\mathrm{PS}(n, w^n, \boldsymbol{s}^n, \boldsymbol{s}^{-n}, \triangle s)$
2: **repeat**
3: $\mathcal{K}_n^{pos} \leftarrow \{k : s_k^n \geq \triangle s\}$.
4: **for all** $k' \in \mathcal{K}_n^{pos}$ **do**
5: $\tilde{\boldsymbol{s}}^n \leftarrow \boldsymbol{s}^n$
6: $\tilde{s}_k^n \leftarrow \tilde{s}_k^n - \triangle s$
7: $\triangle J_-^n(k') \leftarrow J^n(\tilde{\boldsymbol{s}}^n) - J^n(\boldsymbol{s}^n)$
8: **end for**
9: $k_-^{\mathrm{opt}} = \arg\max_{k'} \triangle J_-^n(k')$
10: $s_{k_-^{\mathrm{opt}}}^n \leftarrow s_{k_-^{\mathrm{opt}}}^n - \triangle s$
11: **for all** $k' \in \mathcal{K}$ **do**
12: $\tilde{\boldsymbol{s}}^n \leftarrow \boldsymbol{s}^n$
13: $\tilde{s}_k^n \leftarrow \tilde{s}_k^n + \triangle s$
14: $\triangle J_+^n(k') \leftarrow J^n(\tilde{\boldsymbol{s}}^n) - J^n(\boldsymbol{s}^n)$
15: **end for**
16: $k_a^{\mathrm{opt}} = \arg\max_{k'} \triangle J_+^n(k')$
17: $s_{k_+^{\mathrm{opt}}}^n \leftarrow s_{k_+^{\mathrm{opt}}}^n + \triangle s$
18: $\triangle J^n = \triangle J_-^n(k_-^{\mathrm{opt}}) + \triangle J_+^n(k_+^{\mathrm{opt}})$
19: **until** $\triangle J^n = 0$
20: **return** $\boldsymbol{s}^n$
21: **end procedure**

The PS subroutine is specified in Algorithm 3. Line 3 finds the set of tones on which a decrease of PSD will not lead to a negative PSD. Lines 4 to 10 perform the subtraction phase, and lines 11 to 17 perform the addition phase. If a spectral mask constraint is applied, then in the addition phase we exclude from consideration any tones where addition would result in a spectral mask violation. Each user n always achieves a better objective $J^n(\boldsymbol{s}^n)$ at the end of the PS subroutine, compared with the one achieved by using ASB-S1 algorithm before the PS subroutine. This is due to the monotonic increase of $J^n(\boldsymbol{s}^n)$ during the iterations of the subroutine. Therefore, it is clear that the following is true.

Proposition 1: The PS subroutine always converges.

The convergence of the ASB-A1 algorithm is difficult to show in general, due to the nonconvexity of Problem (OPT2) and the fact that the PS subroutine can only reach a local optimal solution. In our simulations, however, the ASB-A1 algorithm always converges.

Note also that, at the end of each iteration of the PS subroutine, the power constraint of user n is always tight. This is because we take $\triangle s$ away from one tone in the subtraction phase, and put it back into one tone in the addition phase. Thus the resource is always fully utilized and no power violation occurs. This is different from the bit-addition and bit-subtraction algorithms in [10], where the power constraints are either loose or violated during the whole process of the algorithm before convergence.

VI. Convergence Analysis

In this section we prove convergence for various versions of ASB. We will only consider the *rate adaptive (RA)* mode, where users fix their weights $\boldsymbol{w}$ and aim at maximizing their rates under a total power constraint [14].[7] We notice that previous DSL literature (e.g., [3], [5]–[8], [10], [11]) also focus on the RA mode when discussing convergence. Even when $\boldsymbol{w}$ adapts to enforce target rate constraints, extensive simulations show that the algorithms proposed in this paper still converge.

We first discuss the convergence of ASB-S1 in a two-user case. The convergence of ASB-A1 has been briefly mentioned in Proposition 1 for PS subroutine. We then consider the high signal-to-noise ratio (SNR) regime for the reference line, under which we prove stronger convergence results in both the synchronous and asynchronous cases.

A. Convergence of ASB-S1 Algorithm

Here we discuss the convergence of the ASB-S1 algorithm, where the nonconvexity of (9) makes it difficult to prove convergence. In the two-user case, we can still 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-S1, and the algorithm converges if users start from initial PSD values $(s_k^1, s_k^2) = (0, P^2)$ or $(s_k^1, s_k^2) = (P^1, 0)$ on all tones.

The proof of Theorem 1 uses supermodular game theory [21] and strategy transformation similar to [22], and is omitted here due to space limitation. Supermodular game theory can be used to deal effectively with nonconvexity problems, and the convergence result in Theorem 1 does not require any condition on the crosstalk channels. However, it is only for the case of fixed $\boldsymbol{\lambda}$, and users have to initialize their PSD at particular values.

B. Convergence Under High SNR Regime of the Reference Line

To reduce the computation complexity and gain more insight into the solution structure, we simplify the problem under high SNR approximation of the reference line as shown below.

1) Synchronous Transmission Case: The data rate of the reference line can be written as a linear function of the transmission power of user n under additional assumptions. First, from (4) we know that the reference line's rate $\tilde{b}_k^n$ is a decreasing and concave function in user n's transmission power s_k^n, and we can approximate $\tilde{b}_k^n$ with the following linear lower bound:

$$\begin{aligned}\tilde{b}_k^n(s_k^n) &\approx \tilde{b}_k^n(0) + \left.\frac{\partial \tilde{b}_k^n(s_k^n)}{\partial s_k^n}\right|_{s_k^n=0} \cdot s_k^n \\ &= \log\left(1 + \frac{\tilde{s}_k}{\tilde{\sigma}_k}\right) - \frac{\tilde{\alpha}_k^n}{\tilde{\sigma}_k}\frac{\tilde{s}_k}{\tilde{s}_k + \tilde{\sigma}_k} s_k^n. \end{aligned} \tag{10}$$

In other words, this gives the *upperbound* on the *rate loss* of the reference line due to the interference from user n. Second, if we assume that the reference line operates in the high SNR regime whenever it is active, i.e., if $\tilde{s}_k > 0$ then $\tilde{s}_k \gg \tilde{\sigma}_k$, then (10) can be further simplified as

$$\tilde{b}_k^n(s_k^n) \approx \left(\log\left(\frac{\tilde{s}_k}{\tilde{\sigma}_k}\right) - \frac{\tilde{\alpha}_k^n s_k^n}{\tilde{\sigma}_k}\right)\mathbf{1}_{\{\tilde{s}_k>0\}} \tag{11}$$

[7]The other categories of the spectrum balancing operation include *Fixed Margin* (FM) mode and *Margin Adaptive* (MA) mode, In FM, users try to minimize their power consumption under a target rate constraint. In MA, the users maximize their margins after achieving the target data rate.

where $\mathbf{1}_{\{\mathfrak{A}\}}$ is the indicator function and equals to one when event $\mathfrak{A}$ is true. Under (11), Problem (OPT2) becomes a convex optimization problem. In particular, user n's maximization objective function on tone k in (7) is approximated by

$$\begin{aligned} &L_k^n\left(w^n,\lambda^n,s_k^n,s_k^{-n}\right)\\ &\quad=(1-\lambda^n)\left(w^n b_k^n-\frac{(1-w^n)\tilde{\alpha}_k^n s_k^n}{\tilde{\sigma}_k}\mathbf{1}_{\{\tilde{s}_k>0\}}\right)\\ &\qquad-\lambda^n s_k^n+(1-\lambda^n)(1-w^n)\log\left(\frac{\tilde{s}_k}{\tilde{\sigma}_k}\right)\mathbf{1}_{\{\tilde{s}_k>0\}} \end{aligned}$$

thus the corresponding optimal PSD can be found in close form as

$$s_k^n\left(w^n,\lambda^n,s_k^{-n}\right)=\left[\frac{w_n(1-\lambda^n)}{\lambda^n+(1-w^n)(1-\lambda^n)\frac{\tilde{\alpha}_k^n}{\tilde{\sigma}_k}\mathbf{1}_{\{\tilde{s}_k>0\}}}-\sum_{m\neq n}\alpha_k^{n,m}s_k^m-\sigma_k^n\right]^+ \tag{12}$$

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*. Solution (12) 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 $\tilde{\alpha}_k^n$ is higher, or the noise level on the reference line $\tilde{\sigma}_k$ is smaller, or there is more interference plus noise $\sum_{m\neq n}\alpha_k^{n,m}s_k^m+\sigma_k^n$ on the current tone.

This leads to a second version of the ASB algorithm in the synchronous case, ASB-S2 algorithm as shown in Algorithm 4.

Algorithm 4: ASB-S2: ASB-S1 under high SNR regime

1: Replace Line 10 in Algorithm 1 with

$$s_k^n\leftarrow\left[\frac{w_n(1-\lambda^n)}{\lambda^n+(1-w^n)(1-\lambda^n)\frac{\tilde{\alpha}_k^n}{\tilde{\sigma}_k}\mathbf{1}_{\{\tilde{s}_k>0\}}}-\sum_{m\neq n}\alpha_k^{n,m}s_k^m-\sigma_k^n\right]^+.$$

The ASB-S2 algorithm turns out to be a special case of the ASB-A2 introduced next for the asynchronous case, of which the convergence results will be presented in Section VI-B-3.

2) Asynchronous Transmission Case: Due to the coupling induced by ICI, it is very difficult to find the global optimal solution of Problem (OPT2) in the asynchronous case. However, if we also assume the high SINR regime and a linear approximation of the bit per tone formula on the reference line as in the synchronous case, we have

$$\begin{aligned}\tilde{b}_k^n&=\log\left(1+\frac{\tilde{s}_k}{\sum_{j=1}^K\gamma(k-j)\tilde{\alpha}_{k,j}^n s_j^n+\tilde{\sigma}_k}\right)\\&\approx\left(\log\left(\frac{\tilde{s}_k}{\tilde{\sigma}_k}\right)-\frac{\sum_j\gamma(k-j)\tilde{\alpha}_{k,j}^n s_j^n}{\tilde{\sigma}_k}\right)\mathbf{1}_{\{\tilde{s}_k>0\}}.\end{aligned}$$

Then, Problem (OPT2) becomes not only convex but also with a objective function that is separable across tones., i.e.,

$$\begin{aligned}J^n(\boldsymbol{s}^n)=\sum_k\Bigg(&w^n b_k^n-(1-w^n)\sum_j\frac{\gamma(j-k)}{\tilde{\sigma}_j}\tilde{\alpha}_{j,k}^n\mathbf{1}_{\{\tilde{s}_j>0\}}s_k^n\Bigg)\\&+(1-w^n)\sum_k\log\left(\frac{\tilde{s}_k}{\tilde{\sigma}_k}\right)\mathbf{1}_{\{\tilde{s}_k>0\}}\end{aligned}$$

and the corresponding optimal PSD that solves Problem (OPT2) is given as

$$\begin{aligned}&s_k^n(w^n,\lambda^n,\boldsymbol{s}^{-n})\\&\quad=\left[\frac{w^n(1-\lambda^n)}{\lambda^n+(1-\lambda^n)(1-w^n)\sum_j\frac{\gamma(j-k)}{\tilde{\sigma}_j}\tilde{\alpha}_{j,k}^n\mathbf{1}_{\{\tilde{s}_j>0\}}}-\sum_{m\neq n}\left(\sum_j\gamma(k-j)\alpha_{k,j}^{n,m}s_j^m\right)-\sigma_k^n\right]^+\end{aligned} \tag{13}$$

where λ^n is chosen to make the total power constraint tight $\sum_k s_k^n=P^n$. This is a generalization of the frequency selective waterfilling solution of ASB-S2. The complete ASB-A2 algorithm is given in Algorithm 5.

Algorithm 5: ASB-A2: ASB-A1 under high SNR regime

1: Replace Line 10 in Algorithm 1 with

$$s_k^n\leftarrow\left[\frac{w^n(1-\lambda^n)}{\lambda^n+(1-\lambda^n)(1-w^n)\sum_j\frac{\gamma(j-k)}{\tilde{\sigma}_j}\tilde{\alpha}_{j,k}^n\mathbf{1}_{\{\tilde{s}_j>0\}}}-\sum_{m\neq n}\left(\sum_j\gamma(k-j)\alpha_{k,j}^{n,m}s_j^m\right)-\sigma_k^n\right]^+.$$

3) Convergence of Algorithms ASB-S2/A2: We first consider the convergence in the two-user case where users sequentially optimize their PSD levels.

Theorem 2: The ASB-A2 algorithm globally converges to the unique fixed point in a two-user system under fixed $\boldsymbol{w}$, if $\max_k(\sum_j\gamma(j-k)\alpha_{j,k}^{1,2})\max_k(\sum_j\gamma(j-k)\alpha_{j,k}^{2,1})<1$.

Proof of Theorem 2 is given in Appendix A. The key idea behind the proof is that the ASB-A2 algorithm leads to a contraction mapping in the PSD updates, when the maximum product of the crosstalk channel gains is small enough. One extreme case is in a practical CO/RT mixed deployment case, where the crosstalk from CO to RT is negligible (i.e., $\max_k(\sum_j\gamma(j-k)\alpha_{j,k}^{1,2})\max_k(\sum_j\gamma(j-k)\alpha_{j,k}^{2,1})\ll 1$.).

Corollary 1: The ASB-S2 algorithm globally and geometrically converges to the unique fixed point in a two-user system under fixed $\boldsymbol{w}$, if $\max_k\alpha_k^{2,1}\max_k\alpha_k^{1,2}<1$.

Corollary 1 recovers the convergence results for iterative water-filling in the two-user case [3] as a special case (by deactivating the reference line).

We further extend the convergence results to a system with an arbitrary number $N>2$ of users. We consider both sequential

and parallel PSD updates of the users. In the more realistic but harder-to-analyze parallel updates, time is divided into slots, and each user n updates its PSD simultaneously with other users in each time slot according to (13) based on the PSDs from the previous time slot, and the λ^n is adjusted such that the power constraint is tight.

Theorem 3: Assume $\max_{m\neq n,k}(\sum_j \gamma(j-k)\alpha_{j,k}^{n,m}) < 1/(N-1)$, then the ASB-A2 algorithm globally and geometrically converges to the unique fixed point in an N-user system under fixed $\boldsymbol{w}$, with either sequential or parallel updates.

Proof of Theorem 3 is given in Appendix B. For ASB-S2 algorithm, we have the following.

Corollary 2: If $\max_{m\neq n,k}\alpha_k^{n,m} < 1/(N-1)$, then the ASB-S2 algorithm globally and geometrically converges to the unique fixed point in an N-user system under fixed $\boldsymbol{w}$, with either sequential or parallel updates.

Corollary 2 recovers the convergence results for iterative water-filling in an N-user case with sequential updates (proved in [11]) as a special case. Interestingly, the convergence proof for the parallel updates turns out to be simpler than that for sequential updates. Although convergence proofs are given for the case with a time-invariant update order, in our experience the algorithm always converges even if the modems are updated in a random fashion. Because of this, no centralized coordination is necessary for the updating of the individual modems, and this can instead be triggered locally when a modem notices a change in its local conditions, e.g., an increase in the measured noise spectrum as a new modem comes online.

4) Physical Meaning of Convergence Conditions: The convergence conditions in Theorems 2 and 3 and Corollaries 1 and 2 can be translated into constraints on the DSL network topologies. In downstream ADSL, the constraint can be translated into the maximum distance between the transmitters of RT and the CO, which limits the degree of crosstalk the RT transmitter can generate to CO receiver. In upstream VDSL, this means that lines cannot have lengths that are too different from one another, otherwise the near–far effect from the short lines into the long lines will cause severe crosstalk.

To make the physical meaning more concrete, we consider a detailed DSL channel model that relates the channel gain to the network topology. The magnitude of the *direct channel* can be modeled as $h_k^{n,n} = e^{-\beta_k d}$, where β_k is the line propagation constant, which depends on tone index k, and d is the line length. The value of β_k is well understood, and very accurate models exist based on frequency, and the line diameter, construction, materials, etc. The *crosstalk channel*, on the other hand, is not as well understood.[8] However, worst 1% case models for the crosstalk channel have been developed, with which we can develop bounds that will guarantee convergence in 99% of lines. To be specific, the channel gain from transmitter m to receiver n in the worst 1% case crosstalk model is ([12], [13]) $h_k^{n,m} = K_{fext} l_{\text{coupling}} f_k e^{-\beta_k l_{\text{crosstalk}}}$. The constant $K_{fext} = 10^{-45/20}$, l_{coupling} is the length (in kilometers) over which line m and n come into close contact in the same binder and electromagnetic coupling can occur, f_k is the frequency on tone k (in megahertz), and $l_{\text{crosstalk}}$ is the distance from the transmitter of m to the receiver of line n (in kilometers). A graphic illustration of the notations is shown in Fig. 2.

[8]Significant progress has been made in developing more advanced crosstalk channel models that take interpair distance and twisting imperfections into account [17], however this work requires detailed knowledge of the twisted pair geometry in order to estimate the crosstalk channels, something that is not available in an autonomous algorithm.

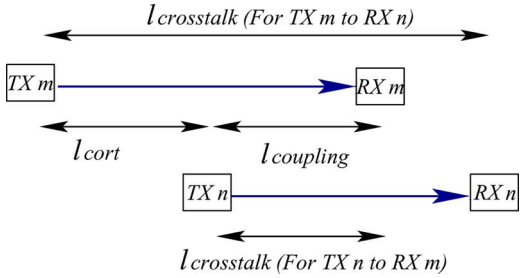


Fig. 2. Physical parameters of the DSL network.

The convergence conditions for ASB-S2 is based on normalized channel gains $\alpha_k^{n,m} = \Gamma h_k^{n,m}/h_k^{n,n}$. First consider the two user downstream ADSL case. For the channel from the CO TX to the RT RX, $l_{\text{crosstalk}} = l_{\text{cort}} + l_{\text{rt}}$, where l_{cort} is the length from the CO TX to the RT TX, and l_{rt} is the length of the RT line. In this case, we have $\alpha_k^{\text{RT,CO}} = K_{fext} l_{\text{coupling}} f_k e^{-\beta_k(l_{\text{cort}}+l_{\text{rt}})}/e^{-\beta_k l_{\text{rt}}} = K_{fext} l_{\text{coupling}} f_k e^{-\beta_k l_{\text{cort}}}$. For ADSL, the maximum deployment length is typically 5 km, so we can use this to bound $l_{\text{coupling}} \leq 5\text{ km} - l_{\text{cort}}$, i.e., $\alpha_k^{\text{RT,CO}} \leq K_{fext}(5 - l_{\text{cort}}) f_k e^{-\beta_k l_{\text{cort}}}$. For any particular value of l_{cort}, the upperbound of $\alpha_k^{\text{RT,CO}}$ can be maximized across k, which is typically achieved at $k = 256$ which corresponds to the highest frequency at 1.1 MHz (i.e., interference is most severe on high frequencies). Next, consider the channel from the RT into the CO, $\alpha_k^{\text{CO,RT}} = K_{fext} l_{\text{coupling}} f_k e^{-\beta_k(l_{\text{CO}}-l_{\text{cort}})}/e^{-\beta_k l_{\text{CO}}} = K_{fext} l_{\text{coupling}} f_k e^{\beta_k l_{\text{cort}}}$, where $l_{\text{coupling}} = l_{\text{CO}} - l_{\text{cort}} \leq 5 - l_{\text{cort}}$. We can again maximize $\alpha_k^{\text{CO,RT}}$ across k (up to 1.1 MHz) for any particular value of l_{cort}. To satisfy the convergence condition in Corollary 1, we need to find l_{cort} such that $\max_k \alpha_k^{\text{CO,RT}} \max_k \alpha_k^{\text{RT,CO}} < 1 = 0\text{ dB}$ in the synchronous case. Using an SNR-gap Γ of 12 dB, which includes a coding gain of 4.2 dB and a noise margin of 6 dB, it turns out that all values of $l_{\text{cort}} \in [0, 5]\text{ km}$ satisfy the convergence conditions as shown in Fig. 3, which means ASB-S2 always converges in the 2-user case for all deployment scenarios. In the N user case, we find that, in the sufficient condition for convergence, the constraint on the maximum distance between the CO TX and RT TX is too loose to be useful in practice. In our experience we find that ASB converges for a broad range of typical scenarios seen in DSL deployments.

VII. Simulation Results

In this section, we show the performance of the ASB algorithms, using a realistic simulator based on empirical channel models developed in standards and used extensively in the industry [12], [13], [16]. In these simulations we use continuous bitloading and do not apply a bitcap. This is done since it results in PSDs that allow a more intuitive interpretation. It is also possible to apply integer bitloading constraints, a bitcap, and PSD

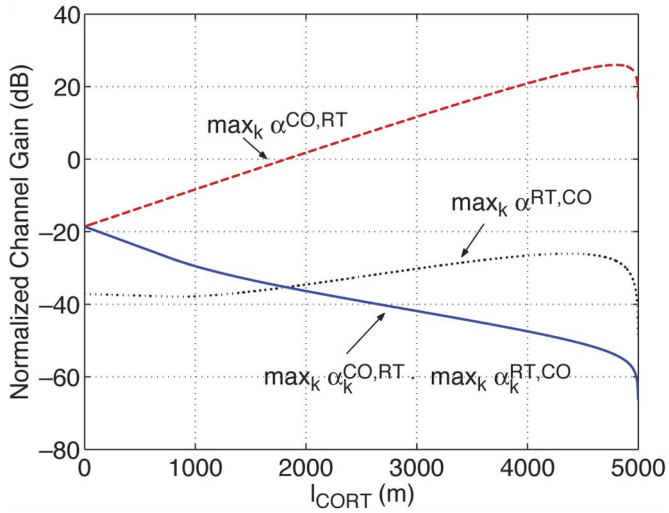


Fig. 3. Convergence conditions always satisfied in the two-user case since $\max_k \alpha_k^{\mathrm{CO,RT}} \max_k \alpha_k^{\mathrm{RT,CO}} < -2.2$ dB.

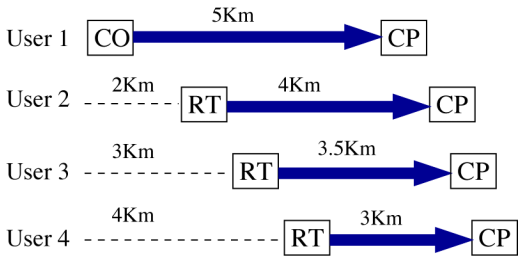


Fig. 4. A four-user mixed CO/RT deployment topology (CP denotes customer premises).

masks to ASB with negligible increase in complexity. We only simulate the performances of the ASB-S1 and ASB-A1 algorithms, which do not involve any high SNR assumptions.

A. Synchronous Transmission Case

Here we summarize a typical numerical example, representative of a large set of experiments we conducted, comparing the performance of the ASB-S1 algorithms with IW, OSB, and ISB in the synchronous transmission case. A four-user mixed CO/RT scenario has been selected to make a comparison with the highly complex OSB algorithm possible. As depicted in Fig. 4, user 1 is CO line, while the other three users are RT lines. ANSI noise model A [23] has been used, which consists of 16 ISDN, 4 HDSL and 10 conventional (non-DSM capable) ADSL disturbers.

Due to the different distances among the corresponding transmitters and receivers, the RT lines generate strong interference into the CO line, while experiencing very little crosstalk from the CO line. The target rates of users 2 and 3 have both been set to 2 Mb/s. User 4 changes its target rate from 0 to 8 Mb/s, and user 1 (the CO line) does not have a target rate constraint and always sets its weight coefficient w^{co} equal to unity in ASB-S1 (i.e., maximizes its own rate without protecting the reference line). The reference line is chosen to match the longest line in the network in terms of background noise and crosstalk channel gains with users in the network. The reference line transmit PSD

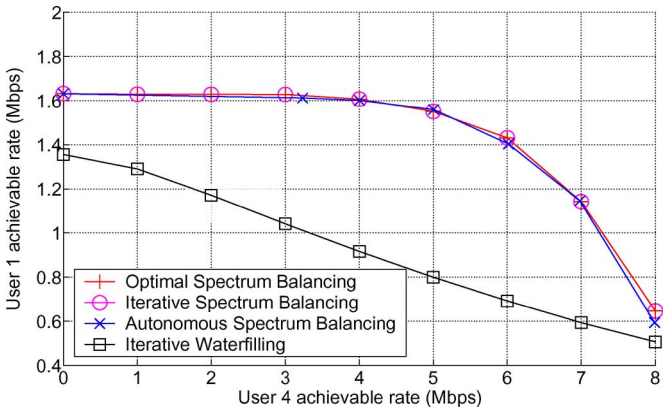


Fig. 5. Rate regions obtained by various DSM algorithms.

is chosen according to single-user waterfilling without considering the interference from other users. Based on this reference line, we get the rate regions[9] shown in Fig. 5. Observe that ASB achieves a near-optimal performance, almost identical to rate regions attained by the globally optimal OSB and ISB, and significant gains over IW. As a typical example, with a target rate of 1 Mb/s on user 1, the rate on user 4 reaches 7.3 Mb/s under ASB algorithm, which is a 143% increase (more than double) compared with the 3 Mb/s achieved by IW.

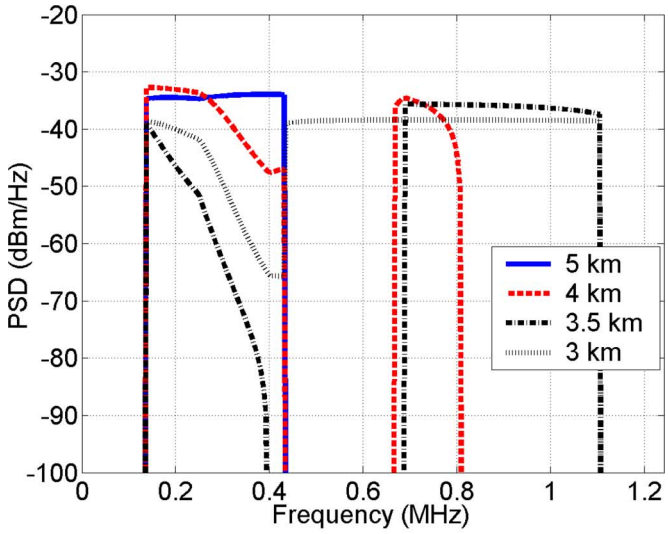
Compared with IW, ASB exploits the special structure of the DSL channel and thus achieves much better performance. Since the direct channel gets worse with increasing frequency and length, long lines cannot effectively utilize high frequencies. The crosstalk channel strength, on the other hand, increases with frequency. In the ASB algorithm, the RT lines transmit with high power in the low frequencies where there is little crosstalk, reduce power in the middle frequencies to protect the reference line, and switch to high power again in the high frequencies where reference line is not active. In the IW algorithm, however, the power allocation is as follows:

$$s_k^n = \left[\frac{w^n(1-\lambda^n)}{\lambda^n} - \sum_{m \neq n} \alpha_k^{n,m} s_k^m - \sigma_k^n \right]^+$$

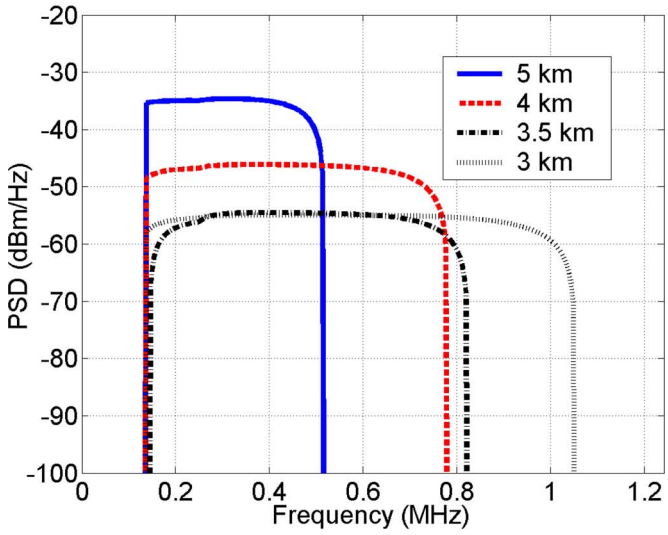
where the adjustable part $w^n(1-\lambda^n)/\lambda^n$ is the *same* on all frequencies. User n first adjusts λ^n such that its total power constraint is tight. If the achieved rate R^n is larger than the target rate $R^{n,\mathrm{target}}$, it performs *equal* power-backoff at all frequencies (i.e., increase the value of λ^n), which unnecessarily reduce the power at the very low (where little crosstalk is generated to the CO line) and high frequencies (where the CO line is inactive). As a result, the IW algorithm leads to highly suboptimal performance, especially in near–far scenarios. As an example, we plot the PSD allocations under the ASB, IW and ISB/OSB algorithms in Fig. 6, with the achievable rates of four users as $R^1 = 1$ Mb/s, $R^2 = R^3 = 2$ Mb/s, $R^4 = 3$ Mb/s for IW and 7.3 Mb/s for ASB-A1/ISB/OSB.

We also summarize a typical simulation of the ASB and IW algorithms in a network with 10 lines, with the line length equal to 5 km for the CO line, and 2 km to 4.5 km for the RTs in

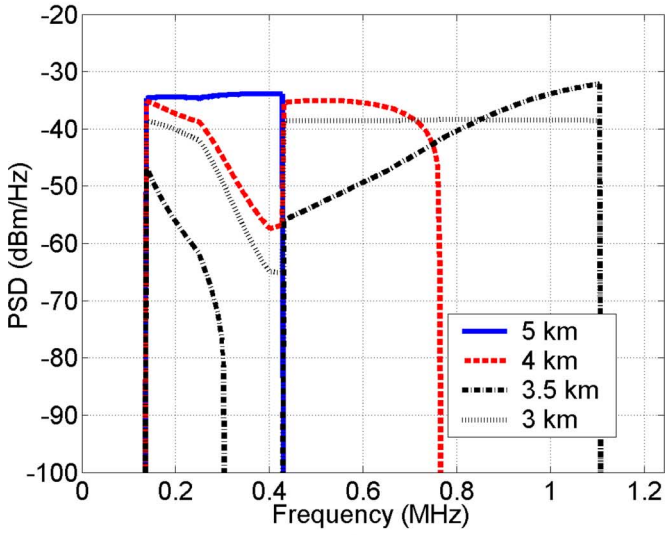
[9]Note that only ASB uses the reference line idea.



(a)



(b)



(c)

Fig. 6. Transmit spectra with synchronous transmission. (a) ASB-S1; (b) IW; (c) ISB/OSB.

steps of 0.3125 km. The RTs are correspondingly located 2 km to 4 km from the CO. The target rate for the CO modem was

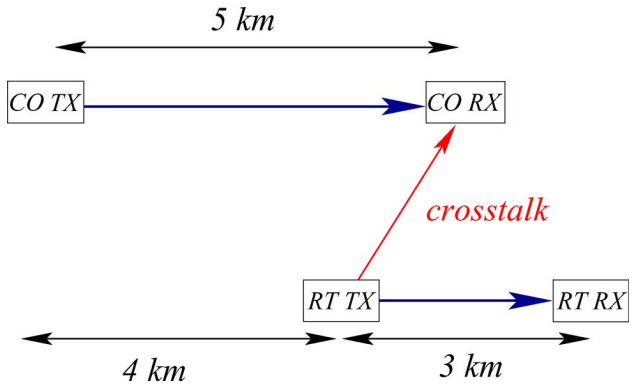


Fig. 7. Mixed CO/RT distribution.

specified as 1.6 Mb/s. With this in mind, the target rates for the RT modems, which are set equally on all RTs, are reduced until the CO modem achieves its target rate. With IW, the RTs are forced to reduce their rates to 0.8 Mb/s in order for the CO to achieve it's target. With ASB, due to the more intelligent allocation of the RT transmit spectra, the RTs can maintain a rate of 2.0 Mb/s while still ensuring that the CO modem achieve 1.6 Mb/s. The ASB algorithm achieves a gain of 122% in the RT rate with respect to IW.

B. Asynchronous Transmission Case

Now consider the case of asynchronous transmission. Here we summarize a typical numerical example comparing the performances of the ASB-A1 and ASB-S1 algorithms. As depicted in Fig. 7, the scenario consists of downstream transmission with two ADSL modems, one 5 km CO line, and one 3 km RT line. The RT TX is deployed 4 km downstream from the CO TX.

Figs. 8 and 9 show an example of the PSDs generated by ASB-A1 and ASB-S1. The target rate for the RT is set to 3.85 Mb/s. Using ASB-S1, which does not take the effects of the ICI into account when optimizing the transmit spectra, the CO achieves 1.3 Mb/s. Using ASB-A1, the CO rate increases to 1.6 Mb/s. With ASB-A1, the transmit power is shifted further into the high-frequencies to prevent excessive ICI to the CO line. Also, since the ICI creates an unavoidable "noise" floor of at around -90 dBm/Hz, it is possible to increase the transmit PSD between 340 KHz and 680 KHz with minimal degradation to the CO line.

Fig. 10 shows the increase in performance relative to IW achieved by ASB-S1 and ASB-A1, respectively, in an asynchronous environment. As we see, even when the modems are not synchronized, ASB-S1 achieves significant gains over IW. Furthermore, if the transmit spectra are further refined through the application of ASB-A1, even further performance gains are possible. For example, if the CO rate is set at 1.4 Mb/s, applying ASB-S1 increases the RT rate by 48% over IW. Applying ASB-A1 leads to a further increase in the RT rate of 186%, leading to a total gain of 234% over IW.

C. Sensitivity Analysis of the Reference Line Choices

In previous simulation examples, we choose the reference line to match the longest line in the binder. Here we study the robustness of the performance to the choice of reference line

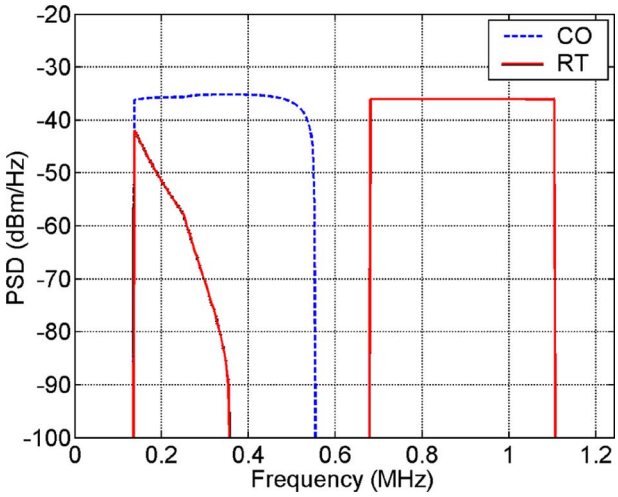


Fig. 8. Transmit spectra with asynchronous transmission: ASB-S1.

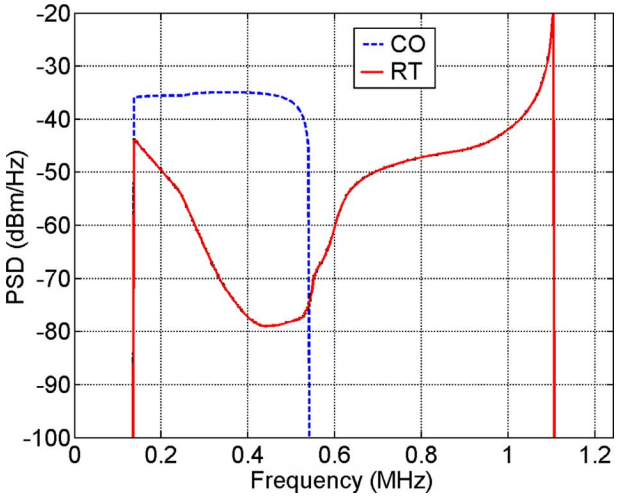


Fig. 9. Transmit spectra with asynchronous transmission: ASB-A1.

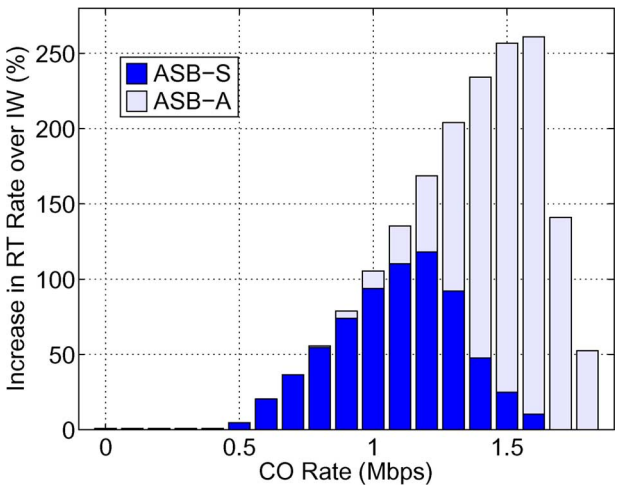


Fig. 10. Performance gains of ASB-S1 and ASB-A1 over IW under asynchronous transmission.

length. We run simulations in a two user scenario as in Fig. 7, assuming that the modems operate synchronously. For these simulations, we vary the length of the reference line, but hold the length of the CO line in the binder at 5 km.

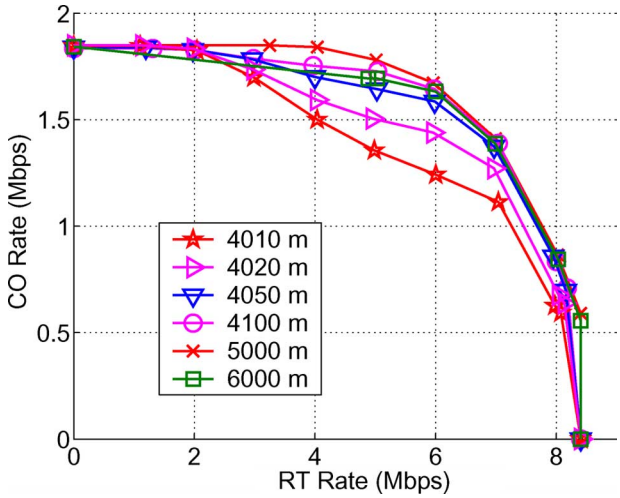


Fig. 11. Sensitivity of ASB-S1 to choice of reference line length.

Fig. 11 shows the achievable rate regions with the different reference line length. Obviously, optimal performance is achieved by setting the reference length to 5000 m, the length of the weaker CO distributed line. We notice that the performance is relatively insensitive to the choice of the reference line length, especially during a broad range of 4050 m to 6000 m. Only when the reference line becomes extremely inaccurate (i.e., around 4020 m or less), which seldom happens in practice, performance starts to degrade rapidly. This is because with a 4020 m reference line, the ASB algorithm assumes that the RT TX is located only 20 m from the reference line RX (recall that the RT RX is actually 4000 m from the CO RX). This will lead to a huge crosstalk channel from the RT to the reference line, and the RT is forced to reduce power in the entire frequency band within which the CO transmits. As can be seen, the performance of ASB is quite insensitive to the mismatch between the length of the reference line and the length of the longest line in the binder. And even for quite big errors in reference line settings, the attainable rate region by ASB is still much larger than IW.

Mathematically, this means that the dependence of the values of the local maxima of this nonconvex optimization problem on the crosstalk channel coefficients is sufficiently insensitive for the observed robustness to hold.

VIII. Complexity Analysis

Here we compare the complexity of ASB-S1 algorithm with the IW algorithm, which is summarized in Table II.[10] *Running time* is measured based on the results of Matlab programs running on an MS-windows machine with a P4-2.8 GHz processor. Real time operations based on hardware implementation would be several orders of magnitude faster. The example we simulated includes a total of $K = 256$ tones and $N = 2$ lines. *Cycles till convergence* is number of outer-cycles required through all of the users before convergence occurs. We typically see that only three outer-cycles are necessary for the rates to converge within 1% of the previous cycle.

[10]The complexity result of ASB-A1 algorithm was summarized in Table I, and the corresponding analysis details are omitted due to space limitation. The complexities of OSB, ISB, and SCALE are too high to be comparable to ASB or IW, and are omitted here.

TABLE II
ALGORITHM COMPLEXITY

Algorithm	Complexity Order	Cycles till conv. (v)	Operations per cycle	Running Time (secs)
IW	$O(NK)$	3	$238NK$	0.01
ASB-S1	$O(NK)$	3	$50864NK$	0.09

A. Complexity Analysis for IW

Iterative waterfilling consists of an outer cycle that iterates through users, and an inner loop that adjusts the total power of the current user until the target data rate is achieved. For each user n, we use a bisection on λ^n within the inner loop, which is both efficient and robust. In the inner loop, each user needs to find the power required to hit its target rate constraint. Typically achieving a precision of 10^{-10} in the total power setting is sufficient to hit the target rate with high accuracy. This requires $\log_2(1/(10^{-10})) = 34$ iterations of bisection search.

For each iteration within the inner loop under a fixed value of λ^n, a standard waterfilling algorithm must be applied with the following complexity[11]:

1) Find the optimal water level such that the total power constraint is satisfied and allocated power is positive on all active tones: $3K$ operations [24].
2) Calculate s_k^n based on the optimal water level: K operations.
3) Calculate the corresponding bitloading: $3K$ operations.

Hence the total complexity of a single waterfilling is $7K$ operations, where one operation is either an addition or a multiplication. Considering the 34 iterations of the bisection search, the iteration through all of the users, and the iteration of the whole process until convergence, the total complexity of IW is then: $v * N * 34 * 7K = 238vNK$, where v is the number of cycles required until convergence.

B. Complexity Analysis for ASB-S1

ASB-S1 consists of three levels of iterations, with the outermost cycle iterating through users. Within each cycle, each users runs an outer loop where it updates w_n until the target data rate is achieved, and an inner loop where it updates λ_n until the total power constraint is satisfied. The bisection search is used in both loops. To achieve a precision of 10^{-10} in both w^n and λ^n, we need a total of $34^2 = 1156$ iterations. Within each iteration, the complexity is dominated by finding the roots of a cubic equation [e.g., solving (9)], which requires 44 operations in total [25]. This has to be repeated on all tones, leading to a total complexity of $44K$. Hence the total complexity of ASB-S1 is $v*N*1156*44K = 50864vNK$. High SNR approximation would further reduce the operations count.

It is important to realize that the order of complexity for ASB is the same as IW: $O(NK)$, and the actual running time of ASB is still well within the bounds for practical implementation. This implementation viability is in sharp contrast to the higher complexity order and centralized schemes of OSB and ISB, which do not offer much rate region gains over ASB.

[11]Also, the inverse Channel-Signal-to-Noise-Ratio (CSNR) must be calculated, and the tones sorted according to the CSNR. However this only needs to be done once for each outer cycle, and can be re-used for all inner-loop iterations. Hence this has a minimal impact on complexity.

IX. CONCLUSION

This paper developed a suite of DSM algorithms referred to as ASB, which are autonomous, have a low complexity and achieves significant performance gains over the prior state-of-the-art autonomous algorithms such as Iterative Waterfilling. In typical scenarios ASB also achieves near-optimal performance, which was previously only possible with the centralized, and highly complex Optimal Spectrum Balancing algorithm.

The convergence of the ASB is proven for an arbitrary number of users in rate-adaptive mode. In particular, ASB includes IW as a special case, thus the convergence proof of our algorithm extends and generalizes the convergence proof of IW. ASB can improve system performance with both synchronous and asynchronous transmission, where the latter is a particularly under-explored research area where only limited, high-complexity heuristics were previously available.

The key concept that enables ASB to successfully tackle the non-convex, coupled, and high-dimensional optimization problem is the reference line, which allows each user to optimize its transmit spectra independently. Each user attempts to achieve its own target rate whilst minimizing the degradation caused to the reference line, which represents a typical victim line within the DSL network. ASB applies this approach of "static pricing" coordination in a rigorous manner with provable theoretical properties, leading to a significantly enlarged rate region compared with IW. The reference line idea can be readily implemented using the reference lines already developed within standards. Although we have focused primarily on ADSL in this paper, ASB is also applicable in VDSL systems.

APPENDIX

A. Proof of Theorem 2

The following Lemma is useful for proving Theorem 2.

Lemma 1: Consider any non-decreasing function $f(x)$ and non-increasing function $g(x)$. If there exists a unique x^* such that $f(x^*) = g(x^*)$, and the functions $f(x)$ and $g(x)$ are strictly increasing and strictly decreasing at $x = x^*$ respectively, then $x^* = \arg\min_x\{\max\{f(x), g(x)\}\}$.

Proof of Lemma 1: For any $\Delta x > 0$, $f(x^* + \Delta x) > f(x^*) = g(x^*) > g(x^* + \Delta x)$. Similarly for any $\Delta x < 0$, $f(x^* + \Delta x) < f(x^*) = g(x^*) < g(x^* + \Delta x)$. It then can be verified that $x^* = \arg\min_x\{\max\{f(x), g(x)\}\}$. ■

Denote $s_k^{n,t}$ as the PSD of user n on tone k after iteration t, where $\sum_k s_k^{n,t} = P^n$ is satisfied at the end of any iteration t for any user n. One iteration is defined as one round of updates of all users. The PSD update in the two-user case can be written as follows:

$$s_k^{n,t+1} = \left[\frac{w^n(1-\lambda^{n,t+1})}{\lambda^{n,t+1} + (1-\lambda^{n,t+1})(1-w^n)\beta_k^n} - \sum_j \gamma(k-j)\alpha_{k,j}^{n,m} s_j^{m,t} - \sigma_k^n\right]^+ \quad (14)$$

where $\beta_k^n = \sum_j(\gamma(j-k)/\tilde{\sigma}_j)\tilde{\alpha}_{j,k}^n \mathbf{1}_{\{\tilde{s}_j>0\}}$, $n,m \in \{1,2\}$, $m \neq n$ and $\forall k,t$, and $[x]^+ = \max(x,0)$. Also define $[x]^- = \max(-x,0)$. Without loss of generality, we assume that the total power constraint is always satisfied at the end of any iteration. In general, the total power constraint needs not to be tight, e.g., when summation of s_k^n (which is determined by (12)) over all tone k is less than the power constraint P^n even when $\lambda^n = 0$. This might happen in the case where w^n is small enough (i.e., user n's target rate is small). However, we can make the power constraint tight in this case by defining an extra "virtual tone". The data rate achieved by user n on the virtual tone is $\epsilon \cdot s_{\text{virtual}}^n$, where ϵ is a very small number and s_{virtual}^n is the PSD allocated to the virtual tone. Furthermore, the reference line is chosen to be inactive on the virtual tone (i.e., $\tilde{s}_{\text{virtual}} = 0$). Now from the perspective of any actual line, loading power on the virtual tone has very small yet positive impact on its own total rate (with very small value ϵ), and has no impact on the reference line's rate. Hence the user will always take any left over power and load onto the virtual tone, and always operate at full power. Then it is clear that

$$\sum_k \left[s_k^{n,t} - s_k^{n,t'}\right]^+ = \sum_k \left[s_k^{n,t} - s_k^{n,t'}\right]^-, \quad \forall n,t,t'. \tag{15}$$

Also define

$$f^{n,t}(x) = \sum_k \left[\left[\frac{w^n(1-x)}{x+(1-x)(1-w^n)\beta_k^n} - \sum_j \gamma(k-j)\alpha_{k,j}^{n,m} s_j^{m,t} - \sigma_k^n\right]^+ - s_k^{n,t}\right]^-$$

and

$$g^{n,t}(x) = \sum_k \left[\left[\frac{w^n(1-x)}{x+(1-x)(1-w^n)\beta_k^n} - \sum_j \gamma(k-j)\alpha_{k,j}^{n,m} s_j^{m,t} - \sigma_k^n\right]^+ - s_k^{n,t}\right]^+$$

where $n,m \in \{1,2\}$, $m \neq n$ and $\forall k,t$. It is clear that $f^{n,t}(x)$ ($g^{n,t}(x)$, respectively) is non-decreasing (non-increasing) in x, and strictly increasing (strictly decreasing) at $x = \lambda^{n,t+1}$ (unless $f^{n,t}(\lambda^{n,t+1}) = g^{n,t}(\lambda^{n,t+1}) = 0$, which means the PSD converges). From (15) we always have $f^{n,t}(\lambda^{n,t+1}) = g^{n,t}(\lambda^{n,t+1})$. Now we can show that

$$\begin{aligned}
&\max\left\{\sum_k\left[s_k^{1,t+1}-s_k^{1,t}\right]^+, \sum_k\left[s_k^{1,t+1}-s_k^{1,t}\right]^-\right\} \\
&= \max\left\{f^{1,t}(\lambda^{1,t+1}), g^{1,t}(\lambda^{1,t+1})\right\} && (16)\\
&\le \max\left\{f^{1,t}(\lambda^{1,t}), g^{1,t}(\lambda^{1,t})\right\} && (17)\\
&\le \max\left\{\sum_k\left[\sum_j \gamma(k-j)\alpha_{k,j}^{1,2}\left(s_j^{2,t}-s_j^{2,t-1}\right)\right]^+ \right. \\
&\qquad \left. \sum_k\left[\sum_j \gamma(k-j)\alpha_{k,j}^{1,2}\left(s_j^{2,t}-s_j^{2,t-1}\right)\right]^-\right\} && (18)\\
&= \max\left\{\sum_j\left[\sum_k \gamma(j-k)\alpha_{j,k}^{1,2}\left(s_k^{2,t}-s_k^{2,t-1}\right)\right]^+ \right. \\
&\qquad \left. \sum_j\left[\sum_k \gamma(j-k)\alpha_{j,k}^{1,2}\left(s_k^{2,t}-s_k^{2,t-1}\right)\right]^-\right\} && (19)\\
&\le \max\left\{\sum_j\sum_k \gamma(j-k)\alpha_{j,k}^{1,2}\left[s_k^{2,t}-s_k^{2,t-1}\right]^+ \right. \\
&\qquad \left. \sum_j\sum_k \gamma(j-k)\alpha_{j,k}^{1,2}\left[s_k^{2,t}-s_k^{2,t-1}\right]^-\right\} && (20)\\
&\le \max_k\left(\sum_j \gamma(j-k)\alpha_{j,k}^{1,2}\right) \\
&\quad \cdot \max\left\{\sum_k\left[s_k^{2,t}-s_k^{2,t-1}\right]^+, \sum_k\left[s_k^{2,t}-s_k^{2,t-1}\right]^-\right\} && (21)\\
&\le \max_k\left(\sum_j \gamma(j-k)\alpha_{j,k}^{1,2}\right)\max_k\left(\sum_j \gamma(j-k)\alpha_{j,k}^{2,1}\right) \\
&\quad \cdot \max\left\{\sum_k\left[s_k^{1,t}-s_k^{1,t-1}\right]^+, \sum_k\left[s_k^{1,t}-s_k^{1,t-1}\right]^-\right\} && (22)\\
&< \max\left\{\sum_k\left[s_k^{1,t}-s_k^{1,t-1}\right]^+, \sum_k\left[s_k^{1,t}-s_k^{1,t-1}\right]^-\right\} && (23)
\end{aligned}$$

where (16) follows from the definition of $f^{n,t}$ and $g^{n,t}$, (17) follows by using Lemma 1 and letting $x = \lambda^{1,t}$, (18) follows from the definition of $f^{n,t}$ and $g^{n,t}$, the expression of $s_k^{1,t}$ in (14), and the fact that $[x^+ - y^+]^+ \le [x-y]^+$ and $[x^+ - y^+]^- \le [x-y]^-$ for any x and y, (19) follows by exchanging indexes k and j, (20) follows by using $[\sum_k x_k y_k]^+ \le \sum_k x_k [y_k]^+$ for all $x_k \ge 0$ and y_k, (21) follows by using the circulant property of γ, i.e., $\sum_j \gamma(j-k) = \sum_j \gamma(j)$, (22) by applying the arguments from (16) to (21) again, and finally (23) follows by the condition in Theorem 2. This shows that the ASB-A2 algorithm is a contraction mapping form an initial PSD values. It can be shown that $\max\{\sum_k [x_k]^+, \sum_k [x_k]^-\}$ is a norm, thus ASB-A2 globally converges to a unique fixed point [26, p. 183]. ■

B. Proof of Theorem 3

We first prove the convergence in the parallel update case. The PSD of user n in tone k after iteration $t+1$ is

$$s_k^{n,t+1} = \left[\frac{w^n(1-\lambda^{n,t+1})}{\lambda^{n,t+1}+(1-\lambda^{n,t+1})(1-w^n)\beta_k^n} - \sum_{m\neq n}\left(\sum_j \gamma(k-j)\alpha_{k,j}^{n,m} s_j^{m,t}\right) - \sigma_k^n\right]^+.$$

The rest of the proof can be obtained similar as in Theorem 2 with the following: (see the equation at the top of the next page).

For the sequential update case, the convergence can be proved by combining Lemma 1 and proof of Theorem 3.4.1 in [11]. First, define $D_{\boldsymbol{s}^t,\boldsymbol{s}^{t'}}(n) = \max\{\sum_k [s_k^{n,t} - s_k^{n,t'}]^+, \sum_k [s_k^{n,t} -$

$$\begin{aligned}
&\max_n \max\left\{\sum_k \left[s_k^{n,t+1}-s_k^{n,t}\right]^+, \sum_k \left[s_k^{n,t+1}-s_k^{n,t}\right]^-\right\} \\
&\le \max_n \max\left\{\sum_j \left[\sum_{m\neq n}\left(\sum_k \gamma(j-k)\alpha_{j,k}^{n,m}\left(s_k^{m,t}-s_k^{m,t-1}\right)\right)\right]^+, \sum_j \left[\sum_{m\neq n}\left(\sum_k \gamma(j-k)\alpha_{j,k}^{n,m}\left(s_k^{m,t}-s_k^{m,t-1}\right)\right)\right]^-\right\} \\
&\le (N-1)\max_n \cdot \max\left\{\max_{m\neq n,k}\left(\sum_j \gamma(j-k)\alpha_{j,k}^{n,m}\right)\sum_k \left[s_k^{m,t}-s_k^{m,t-1}\right]^+, \max_{m\neq n,k}\left(\sum_j \gamma(j-k)\alpha_{j,k}^{n,m}\right)\sum_k \left[s_k^{m,t}-s_k^{m,t-1}\right]^-\right\} \\
&\le (N-1)\max_{m\neq n,k}\left(\sum_j \gamma(j-k)\alpha_{j,k}^{n,m}\right)\cdot \max_n \max\left\{\sum_k \left[s_k^{m,t}-s_k^{m,t-1}\right]^+, \sum_k \left[s_k^{m,t}-s_k^{m,t-1}\right]^-\right\} \\
&< \max_n \max\left\{\sum_k \left[s_k^{m,t}-s_k^{m,t-1}\right]^+, \sum_k \left[s_k^{m,t}-s_k^{m,t-1}\right]^-\right\}.
\end{aligned}$$

$s_k^{n,t'}]^-\}$, and $\boldsymbol{D}_{\boldsymbol{s}^t,\boldsymbol{s}^{t'}} = \{D_{\boldsymbol{s}^t,\boldsymbol{s}^{t'}}(n), \forall n\}$. Using induction, we can find an $N \times N$ matrix $\boldsymbol{H}$ such that $\boldsymbol{D}_{\boldsymbol{s}^{t+1},\boldsymbol{s}^t} \le \boldsymbol{H}\boldsymbol{D}_{\boldsymbol{s}^t,\boldsymbol{s}^{t-1}}$. The final step is to show the maximum eigenvalue of matrix $\boldsymbol{H}$ is less than 1, which guarantees that ASB-A2 algorithm is an contraction mapping in the sequential updates. Details are omitted due to space limitations. ■

ACKNOWLEDGMENT

The authors would like to acknowledge helpful discussions from J. Cioffi, A. Fraser, A. H. Mohsenian-Rad, D. Palomar, C. W. Tan, W. Yu, and J. (Steve) Yuan.

REFERENCES

[1] R. Cendrillon, G. Ginis, and M. Moonen, "Improved linear crosstalk precompensation for downstream VDSL," in *Proc. IEEE Int. Conf. Acoust., Speed Signal Process. (ICASSP)*, 2004, pp. 1053–1056.

[2] G. Ginis and J. M. Cioffi, "Vectored transmission for digital subscriber line systems," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 5, pp. 1085–1104, 2002.

[3] W. Yu, G. Ginis, and J. Cioffi, "Distributed multiuser power control for digital subscriber lines," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 5, pp. 1105–1115, Jun. 2002.

[4] Z.-Q. Luo and J.-S. Pang, "Analysis of iterative waterfilling algorithm for multiuser power control in digital subscriber lines," *EURASIP J. Appl. Signal Process.*, 2006, vol. 2006, Article ID 24012, pp. 1"C10.

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

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

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

[8] J. Papandriopoulos and J. Evans, "Low-complexity distributed algorithms for spectrum balancing in multi-user DSL networks," in *IEEE ICC*, 2006.

[9] J. M. Cioffi, W. Rhee, M. Mohseni, and M. H. Brady, "Band preference in dynamic spectrum manag," in *EUSIPCO Conf. Signal Process.*, Vienna, Austria, Sep. 2004.

[10] V. M. K. Chan, "Multiuser detection and spectrum balancing for digital subscriber lines," Master's thesis, Dep. Elect. Comp. Eng., Univ. Toronto, Toronto, 2005.

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

[12] *Very High Speed Digital Subscriber Line (VDSL); Functional Requirements, Rev. V.1.3.1*, ETSI Std. TS 101 270-1, 2003.

[13] *Very-High Bit-Rate Digital Subscriber Lines (VDSL) Metallic Interface*, ANSI Std. T1.424, 2004.

[14] T. Starr, J. Cioffi, and P. Silverman, *Understanding Digital Subscriber Line Technology*. Englewood Cliffs, NJ: Prentice-Hall, 1999.

[15] F. Sjoberg, M. Isaksson, R. Nilsson, and P. Borjesson, "Zipper: A duplex method for VDSL based on DMT," *IEEE Trans. Commun.*, vol. 47, no. 8, p. 1245, 1999.

[16] *Very High Speed Digital Subscriber Line Transceivers 2*, ITU Draft Std. G.993.2, 2006.

[17] B. Lee, "Binder MIMO channels," Ph.D. dissertation, Stanford Univ., Stanford, CA, Nov. 2004.

[18] S. Boyd and L. Vanderberghe, *Convex Optimization*. Cambridge, MA: Cambridge Univ. Press, 2004.

[19] *Spectrum Management for Loop Transmission Systems, Issue 2*, ANSI Std. T1.417, 2003.

[20] D. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientific, 1999.

[21] D. M. Topkis, *Supermodularity and Complementarity*. Princeton, NJ: Princeton Univ. Press, 1998.

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

[23] *Noise Models for VDSL Performance Verification*, ANSI-77E7.4/99. 438R2, Dec. 1999.

[24] J. Cioffi, Advanced Digital Communication EE379C Course Reader, Chapter 4—Multi-Channel Modulation [Online]. Available: http://www.stanford.edu/class/ee379c/readerfiles/chap4.pdf

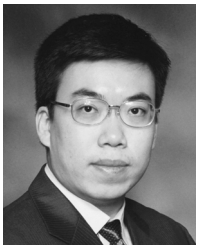
[25] R. Nickalls, "A new approach to solving the cubic: Cardan's solution revealed," *Mathematical Gazette*, vol. 77, pp. 354–359, 1993.

[26] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

Raphael Cendrillon (S'02–M'04) was born in Melbourne, Australia, in 1978. He received the Electrical Engineering (First Class Hons.) degree from the University of Queensland, Australia, in 1999 and the Ph.D. degree in electrical engineering (*summa cum laude*) from the Katholieke Universiteit Leuven (K.U. Leuven), Belgium, in 2004.

In 2002, he was a Visiting Scholar with the Information Systems Laboratory, Stanford University, Stanford, CA, with Prof. J. Cioffi. In 2005, he was a Postdoctoral Research Fellow with the University of Queensland, Australia. During this period, he was also a Visiting Research Fellow with Princeton University, Princeton, NJ, with Prof. M. Chiang. Currently, he is a Senior DSP Engineer with Marvell Hong Kong, Ltd.

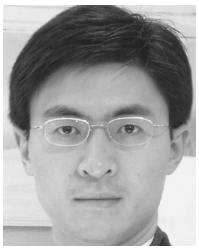
Dr. Cendrillon was awarded the Alcatel Bell Scientific Prize in 2004, the IEEE Travel Grants in 2003, 2004, and 2005, the K.U. Leuven Bursary for Advanced Foreign Scholars in 2004, and the UniQuest Trailblazer Prize for Commercialization in 2005.



Jianwei Huang (S'00–M'06) received the B.S. degree in electrical engineering from Southeast University, Nanjing, China, in 2000 and the M.S. and Ph.D. degrees in electrical and computer engineering from Northwestern University, Evanston, IL, in 2003 and 2005, respectively.

He is a Postdoctoral Research Associate with the Department of Electrical Engineering, Princeton University, Princeton, NJ. In 2004 and 2005, he worked in the Network Advanced Technology Group at Motorola, both as a full-time summer intern and a part-time researcher. In 1999, he worked as a summer intern with the Department of Change Management, GKN Westland Aerospace Co., Ltd. His main research interests lie in the area of communications and networking, with specific areas including cognitive radio networks, wideband OFDM and CDMA systems, wireless medium access control, multimedia communications, cooperative communications, and wired DSL broadband access networks.

Dr. Huang is an Associate Editor of the (Elsevier) *Journal of Computer & Electrical Engineering*, the Lead Guest Editor of the "Game Theory in Communication Systems" special issue of the IEEE JOURNAL OF SELECTED AREAS IN COMMUNICATIONS, the Lead Guest Editor of the "Collaboration and Optimization in Multimedia Communications" special issue of the *Journal of Advances in Multimedia*, and a Guest Editor of the "Cross-Layer Optimized Wireless Multimedia Communications" special issue of the *Journal of Advances in Multimedia*. He is the recipient of a 2001 Walter P. Murphy Fellowship at Northwestern University and a 1999 Chinese National Excellent Student Award.



Mung Chiang (S'00–M'03) received the B.S. (Honors) degree in electrical engineering and mathematics, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 1999, 2000, and 2003, respectively.

He is an Assistant Professor of Electrical Engineering and an affiliated faculty member of the Program of Applied and Computational Mathematics and of the Computer Science Department, Princeton University, Princeton, NJ. He conducts research in the areas of optimization of communication systems, theoretical foundation of network architectures, algorithms in broadband access networks, and stochastic models of communications.

Prof. Chiang has been awarded a Hertz Foundation Fellow, and received the Stanford University School of Engineering Terman Award, SBC Communications New Technology Introduction Contribution Award, NSF CAREER Award, an ONR Young Investigator Award, and the Princeton University Howard B. Wentz Junior Faculty Award. His January 2005 IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS paper became the Fast Breaking Paper in Computer Science in 2006 according to ISI's citation frequency. He coauthored the Best Student Paper at IEEE Globecom 2006. He is the Lead Guest Editor of the Special Issue of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS on "Nonlinear Optimization of Communication Systems," a Guest Editor of the Joint Special Issue of the IEEE TRANSACTIONS ON INFORMATION THEORY and the IEEE/ACM TRANSACTIONS ON NETWORKING on "Networking and Information Theory," an Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, a Program Co-Chair of the 38th Conference on Information Sciences and Systems, and a coeditor of the new Springer book series *Optimization and Control of Communication Systems*.

Marc Moonen (M'94–SM'06–F'07) received the electrical engineering degree and the Ph.D. degree in applied sciences from Katholieke Universiteit Leuven (K.U.Leuven), Belgium, in 1986 and 1990, respectively.

Since 2004, he has been a Full Professor at the Electrical Engineering Department of Katholieke Universiteit Leuven, where he is currently heading a research team of 16 Ph.D. candidates and postdocs, working in the area of numerical algorithms and signal processing for digital communications, wireless communications, DSL, and audio signal processing.

Dr. Moonen received the 1994 K.U.Leuven Research Council Award, the 1997 Alcatel Bell (Belgium) Award (with Piet Vandaele), and the 2004 Alcatel Bell (Belgium) Award (with Raphael Cendrillon) and was a 1997 "Laureate of the Belgium Royal Academy of Science." He received a journal Best Paper Award (with G. Leus) from the IEEE TRANSACTIONS ON SIGNAL PROCESSING and (with S. Doclo) from Elsevier Signal Processing. He was Chairman of the IEEE Benelux Signal Processing Chapter from 1998 to 2002 and has been a European Association for Signal, Speech and Image Processing (EURASIP) AdCom Member since 2000 and a member of the IEEE Signal Processing Society Technical Committee on Signal Processing for Communications. He has served as Editor-in-Chief for the *EURASIP Journal on Applied Signal Processing* from 2003 to 2005 and has been a member of the editorial board of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I from 2002 to 2003 and IEEE SIGNAL PROCESSING MAGAZINE from 2003 to 2005. He is currently a member of the editorial board of *Integration, the VLSI Journal*, *EURASIP Journal on Applied Signal Processing*, *EURASIP Journal on Wireless Communications and Networking*, and *Signal Processing*.