Autonomous Spectrum Balancing for Digital Subscriber Lines

Raphael Cendrillon, Jianwei Huang, Mung Chiang, Marc Moonen

Abstract— **The main performance bottleneck of modern Digital Subscriber Line (DSL) networks is the crosstalk among different lines (users). By deploying Dynamic Spectrum Management (DSM) techniques and reducing excess crosstalks 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), for both synchronous and asynchronous transmission cases. In the synchronous case, the transmissions over different tones are orthogonal to each other. In the asynchronous case, the transmissions on different tones are coupled together due to inter-carrier-interference. In both cases, ASB utilizes the concept of a "reference line", which mimics a typical victim line in the interference channel. The basic procedure in ASB algorithms is simple: each user optimizes the weighted sum of the achievable rates on its own line and the reference line while assuming the interferences from other users as noise. Users then iterate until the target rate constraints are met. Good choices of reference line parameters are already available in industry standards, and the ASB algorithm makes the intuitions completely rigorous and theoretically sound. ASB is the first set of algorithms that is fully autonomous, has low complexity, and yet achieves near-optimal performance. It effectively solves the nonconvex and coupled optimization problem of DSL spectrum management, and overcomes the bottleneck of all previous DSM algorithms.**

Index Terms— **Digital Subscriber Lines, spectrum management, power allocation, distributed algorithm, dual decomposition**

EDICS: SPC-TDLS, SPC-MULT

I. Introduction

A. Motivation

Digital Subscriber Line (DSL) technologies transform traditional voice-band copper channels into broadband access systems, which are capable of delivering data rates up to several Mbps per twisted-pair over a distance of 10 kft in the basic Asymmetric DSL (ADSL). Despite over 140 million DSL lines worldwide as of 2005, 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., [2], [3]) are infeasible in many cases, due to the complexity issues (both amount of computation needed and the requirements of new chip sets) or as a result of unbundling (i.e., incumbent service providers must rent certain lines to their competitors).

Recently, various dynamic spectrum management (DSM) 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 the first set of algorithms that is both autonomous (distributed algorithm across the users without explicit information exchange) with low complexity, provably convergent, and achieving close to the globally optimal rate region in practice.

B. Related Work on DSM Algorithms

One of the first and most well known DSM algorithms is the *Iterative Water-filling* (IW) algorithm [4], where each line maximizes its own data rate by waterfilling over the noise and interference from other lines. The IW algorithm is a completely autonomous algorithm with a linear complexity in the number of users. 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 details in Section II), such as mixed central office and remote terminal deployments of ADSL and upstream VDSL. This is in part due to the greedy nature of the algorithm.

Recently two optimal but centralized DSM algorithms are 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. Unfortunately OSB has an exponential complexity in the number of users, making it intractable for DSL network with more than 5 lines. As an improvement over the OSB algorithm, ISB is proposed to implement the weighted-rate sum optimization in an iterative fashion over the users. This

R. Cendrillon is with Marvell Hong Kong Ltd., Hong Kong, email: raphael@cendrillon.org. J. Huang and M. Chiang are with Dept. of Electrical Engineering, Princeton University, USA, email: {jianweih,chiangm}@princeton.edu M. Moonen is with Dept. of Electrical Engineering, Katholieke Universiteit Leuven, Belgium, email: moonen@esat.kuleuven.be. Partial and preliminary results have appeared in ISIT 2006.

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, the 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 PSDs for all modems. NMC requires knowledge of the crosstalk channels between all lines and all background noise, which is difficult to obtain due to the large number of lines. Identification and transmission of crosstalk channel measurements back to the NMC are not supported in existing standards either. Also, the operation of NMC requires a lot of overhead, in terms of both bandwidth and infrastructure. Furthermore, the 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 impossible 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 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.

In summary, there are currently no DSM algorithms that provide both low complexity, autonomous operation and near-optimal rate region. This paper overcomes this bottleneck through the ASB algorithms.

IW, OSB, ISB, and SCALE mentioned above all assume synchronous transmissions of the modems, which allows crosstalk to be modeled independently on each tone. Unfortunately, this synchronization is almost never 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) significantly complicates the DSM problem further. The state-of-art results for asynchronous transmissions are the two centralized greedy algorithms proposed in [9], bit-subtracting and bit-adding algorithms. Both algorithms start from the power spectrum 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 suite of ASB algorithms proposed in this paper has the following advantages compared with all the previous algorithms. First of all, ASB is autonomous: it can be applied in a distributed fashion across users with no explicitly information exchange. Furthermore, the algorithm has low complexity in both the number of users and tones, and is provably convergent under reasonable conditions on the channel gains that are often satisfied in practice. In the synchronous case, the ASB algorithm has similar complexity as IW, but in the near-far scenario achieves a performance much better than IW and close to ISB and OSB. In the asynchronous case, the ASB algorithm reduces the complexity from those in [9], and achieves significant better performance than the ASB algorithm that does not consider the ICI. 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.

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 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 completely 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. On the other hand, each user 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 previous work on IW [4], [10].

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

Results in this paper hold for any DSL systems topology. To be concrete, we will often examine the typical near-far deployment scenario for downstream ADSL transmissions

TABLE I
COMPARISON OF VARIOUS DSM ALGORITHMS

Algorithm	Operation	Complexity	Performance	Reference
Synchronous Case				
IW	Autonomous	$O(KN)$	Suboptimal	[4]
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 algm.	Centralized	$O\left(N^2K^3\right)$	Suboptimal	[9]
ASB-A1	Autonomous	$O\left(NK^2\log_2(K)\right)$	Suboptimal	this paper

with a frequency band up to 1.1 MHz,[1] as shown in Fig. 1. There are two twisted-pair copper lines in the network. The first line is from the Central Office (CO) to customer 1. Since customer 2 is far away from CO, the service provider deploys a Remote Terminals (RT) near the edge of the network, which connects with customer 2 through a relatively short copper line. In the downstream transmission case shown in the figure, the transmitting modems (TX) are located at the CO and RT, and the receivers (RX) are at the customer homes. Each DSL modem transmits over multiple frequency tones (carriers). Multiple lines sharing the same binder generate crosstalks (interferences) to each other on all frequency tones. Although RT extends the footprint of the DSL network, it also generates excessive interference to the CO line due to the physical proximity between the RT TX and the CO RX. However, CO TX generates little crosstalk to RT RX due to the long distance between them.

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. 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 a mixed VDSL/ADSL deployments, RT-deployed VDSL will damage the CO-deployed ADSL signals in the downstream.

Next we formally introduce the mathematical models for both synchronous and asynchronous transmission cases, following the notation in [5], [6], [9]. More details are given in Section VI-B.4.

A. Synchronous Transmission

We consider a DSL network with a set $\mathcal{N} = \{1, ..., N\}$ users (i.e., lines, transmitting modems) and $\mathcal{K} = \{1, ..., K\}$ 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:

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

[1]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.

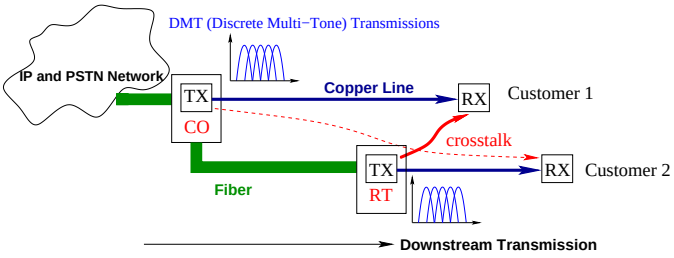


Fig. 1. Mixed CO/RT distribution

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 Spectrum Density (PSD) $s_k^n \triangleq \mathcal{E}\left\{|x_k^n|^2\right\}$, where $\mathcal{E}\{\cdot\}$ denotes expected value. The vector containing the PSD of user n on all tones as $\boldsymbol{s}^n \triangleq \{s_k^n, k \in \mathcal{K}\}$.

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

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

where $\alpha_k^{n,m} \triangleq |h_k^{n,m}|^2 / |h_k^{n,n}|^2$ is the normalized crosstalk channel gain, and $\sigma_k^n \triangleq \mathcal{E}\left\{|z_k^n|^2\right\} / |h_k^{n,n}|^2$ 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 [11]. 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: $\sum_{k\in\mathcal{K}} s_k^n \leq P^n$. 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, $\mathcal{E}\left\{|y_k^n|^2\right\}$, also depends on other users' transmission PSD on tones *other than* 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 [9],

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

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)

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

All the other system parameters and constraints are the same as the synchronous case.

III. Spectrum Management Problem and a General Framework of ASB

The spectrum management problem is defined as follows,

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

Here $R^{n,\text{target}}$ denote 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 to solve. However, the rate region achieved by all users is convex in the asymptomatic case when number of tones becomes large [5]. Thus by changing the values of $R^{n,\text{target}}$ of all users $n>1$, the solution 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 a reference line, a virtual line that represents a typicalvictim user within the DSL system. It turns out that it is adequate to make the reference line correspond to the longest line in the network (e.g. the CO line in a mixed CO/RT scenario in Section VII), which has the weakest direct channel and receives relatively stronger crosstalk from other users. 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.

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 $R^{n,ref}\triangleq\sum_{k\in\mathcal{K}}\tilde{b}_k^n$, 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

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

The coefficients $\{\tilde{s}_k,\tilde{\sigma}_k,\tilde{\alpha}_k^n,\forall k,n\}$ are parameters of the reference line and can be readily obtained from long-term field measurements. Since the crosstalk channel can be regarded as time-invariant in DSL systems, the parameters of the reference lines are known to users a priori. 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:

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

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 ($R^{n,ref}$). 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., [4]–[7], [9]).

To facilitate the analysis in the following sections, we also consider another 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.,

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

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.[2] In the

[2]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 ∞.

synchronous case, it has been shown in [5] that the rate region of Problem (OPT1) (in terms of R^n and $R^{n,ref}$) 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 Problem (OPT1) (i.e., find a w^n such that the solution of Problem (OPT2) satisfies $R^n = R^{n,\text{target}}$) 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.

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). Since the transmissions on different tones are orthogonal to each other here, we can use dual decomposition 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 solving the dual problem can lead to optimal primal solution.[3]

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),

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

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

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

where $\boldsymbol{s}^{-n} = \left(\boldsymbol{s}_k^1, ..., \boldsymbol{s}_k^{n-1}, \boldsymbol{s}_k^{n+1}, ..., \boldsymbol{s}_k^N\right)$ 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

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

[3]Recent results in [12] show that the duality gap is already approximately zero when there are only 8 tones, and the actual number of tones in the current DSL standards is 2 to 3 orders of magnitude larger.

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

where $s_k^{-n} = \left(s_k^1, ..., s_k^{n-1}, s_k^{n+1}, ..., s_k^N\right)$. 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

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

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 $s_k^{n,S1}$.

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 1.

Remark 1: 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 near-optimal performance. Like IW, ASB uses an iterative approach, optimizing the PSD of each user in turn.

Remark 2: 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 [13]–[15]. The reference PSD method specifies that the received PSD of a modem must match the so-called "reference PSD". In VDSL standard, different reference PSDs are defined for different transmission bands (e.g., two bands in VDSL1 and three bands in VDSL2). The referenced PSD is defined as the received PSD of a "typical line" that would be operating in each particular band. The length of this typical line determines its channel attenuation, and the corresponding received PSD. 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 [16]. While in industry standards the use of a reference line was motivated by engineering intuitions and ad hoc developments, here we apply the concept of the reference line from a theoretical foundation. We formulate Problems (OPT1) and (OPT2) based on the reference line idea, and solve the problems based on dual decomposition techniques and optimization theory. This leads to the near-optimal performance of the ASB algorithm. For example, in a two-user network, if the reference line is set to the longest line in the network, the ASB algorithm achieves a performance similar as the optimal but centralized ISB and OSB algorithms. Simulations in Section VII show that the performance of the ASB algorithm is insensitive to inaccuracies or variations in the choice of the reference line parameters.

Remark 3: In considering only a single reference line, the ASB algorithm makes an implicit assumption that, by protecting the weakest 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 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

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

For notational simplicity, we ignore the dependence of J^n on $\boldsymbol{s}^{-n}$ (which is assumed to be fixed during user n's PSD optimization). Now, define $\triangle s$ as the incremental amount of power a user can change on a tone at a time. In other words, $\triangle s$ defines 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 $J^n\left(\boldsymbol{s}^n\right)$ (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 $J^n\left(\boldsymbol{s}^n\right)$ (or smallest decrease similar as in the subtraction phase). This iteration repeats until the *net change* of $J^n\left(\boldsymbol{s}^n\right)$ 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 [17], 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 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** $\triangle s = 0, -1, ..., -100$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** 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: $k_-^{\text{opt}} = \arg\max_{k'} \triangle J_-^n(k')$
10: $s_{k_-^{\text{opt}}}^n \leftarrow s_{k_-^{\text{opt}}}^n - \triangle s$
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: $k_a^{\text{opt}} = \arg\max_{k'} \triangle J_+^n(k')$
17: $s_{k_+^{\text{opt}}}^n \leftarrow s_{k_+^{\text{opt}}}^n + \triangle s$
18: $\triangle J^n = \triangle J_-^n(k_-^{\text{opt}}) + \triangle J_+^n(k_+^{\text{opt}})$
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. Since the value of $J^n(\boldsymbol{s}^n)$ increases in each iteration and is upper-bounded, it must converge. 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 simulation, however, the ASB-A1 algorithm always converges.

Remark 4: 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 to 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 [9], where the power constraints are either loose or violated during the whole process of the algorithm before convergence.

Remark 5: 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.

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 [11].[4] We notice that all previous DSL literature (e.g., [4]–[10]) also focus on the RA mode when discussing convergence.It is worth noting that extensive simulations show that all algorithms proposed in this paper always converge, even when $\boldsymbol{w}$ adapts to enforce target rate constraints.

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 synchronous and asynchronous cases.

A. Convergence of ASB-S1 Algorithm

Here we discuss the convergence of ASB-S1 algorithm, where the nonconvexity of (9) makes it difficult to prove the 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 [18] and strategy transformation similar to [19], 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.

[4]The second main category of the spectrum balancing operation is *Fixed Margin* (FM) mode, where users try to minimize their power consumption under a minimum target rate constraint.

1) Synchronous Transmission Case: The reference line rate 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:

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

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

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

where $\mathbf{1}_{\{\mathfrak{A}\}}$ is the indictor function and equals to one when event $\mathfrak{A}$ is true. Under (11), Problem (OPT2) becomes a convex optimization problem. Especially, 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

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

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.

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 Sect. 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 high SINR regime on the reference line as in the synchronous case, we have

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

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

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

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

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

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.

3) Convergence of Algorithms ASB-S2/A2: We first consider the convergence in a 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 $\max_k\alpha_k^{1,2}\max_k\alpha_k^{2,1} < 1/\left(\sum_k\gamma(k)\right)^2$.

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

1: Replace Line 10 in Algorithm 1 with

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

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

1: Replace Line 10 in Algorithm 1 with

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

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., $\max_k \left\{\alpha_k^{1,2}\right\} \max_k \left\{\alpha_k^{2,1}\right\} \ll 1$). We note that the value of $\sum_k \gamma(k)$ is around 1.66 for a wide range of K (i.e., $32 \leq K \leq 4096$).

It is straightforward to show the following result for ASB-S2.

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 [4] as a special case (by letting the reference line to be inactive).

We further extend the convergence results to a system with an arbitrary $N > 2$ 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 $\max_{m\neq n,k} \alpha_k^{n,m} < \frac{1}{(N-1)\sum_k \gamma(k)}$, 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

Corollary 2: If $\max_{m\neq n,k} \alpha_k^{n,m} < \frac{1}{N-1}$, 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 [10]) as a special case. Interestingly, the convergence proof for the parallel updates turns out to be simpler than that for sequential updates.

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, let us consider a detailed DSL channel model that relates the channel gain to the network topology. The *direct channel* can be modeled $h_k^{n,n} = e^{-\beta_k d}$, 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,

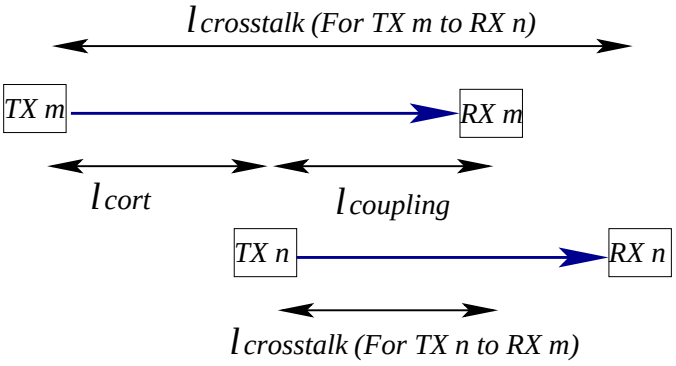


Fig. 2. Physical parameters of the DSL network

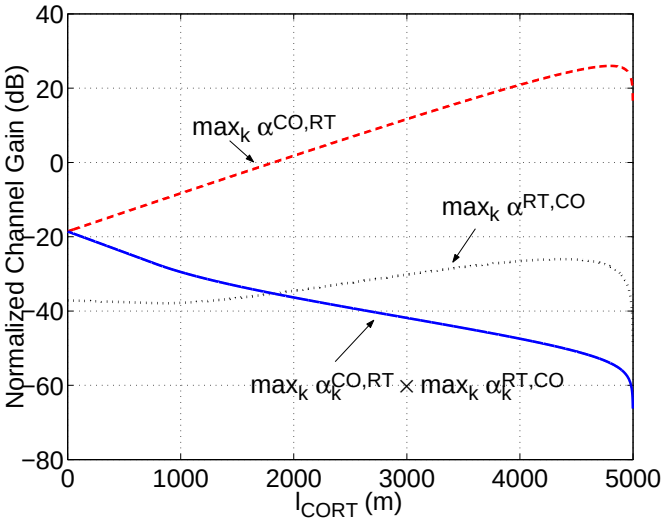


Fig. 3. Convergence conditions always satisfied in the two-user case since $\max_k \alpha_k^{CO,RT} \max_k \alpha_k^{RT,CO} < -2.2$dB.

materials, etc. The *crosstalk channel*, on the other hand, is not as well understood. 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 ([13], [14]) $h_k^{n,m} = K_{fext} l_{coupling} f_k e^{-\beta_k l_{crosstalk}}$. Here constant $K_{fext} = 10^{-45/20}$, $l_{coupling}$ is the length (in km) over which line m and n come into close contact and electromagnetic coupling can occur, f_k is the frequency on tone k (in MHz), and $l_{crosstalk}$ is the distance from the transmitter of m to the receiver of line n (in km). An graphic illustration of the notations is shown in Fig. 2.

The convergence conditions for ASB-S2/A2 are based on normalized channel gains $\alpha_k^{n,m} = h_k^{n,m}/h_k^{n,n}$. First consider the 2 user downstream ADSL case. For the channel from the CO TX to the RT RX, $l_{crosstalk} = l_{cort} + l_{rt}$, 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 $\alpha_k^{RT,CO} = K_{fext} l_{coupling} f_k e^{-\beta_k (l_{cort}+l_{rt})} / e^{-\beta_k l_{rt}} = K_{fext} l_{coupling} f_k e^{-\beta_k l_{cort}}$. For ADSL, the maximum deployment length is typically 5 km, so we can use this to bound $l_{coupling} \leq 5\text{km} - l_{cort}$, i.e., $\alpha_k^{RT,CO} \leq K_{fext}(5 -$

$l_{cort})f_k e^{-\beta_k l_{cort}}$. For any particular value of l_{cort}, the upperbound of $\alpha_k^{RT,CO}$ 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, $\alpha_k^{CO,RT} = K_{fext} l_{coupling} f_k e^{-\beta_k (l_{co} - l_{cort})} / e^{-\beta_k l_{co}} = K_{fext} l_{coupling} f_k e^{\beta_k l_{cort}}$, where $l_{couping} = l_{co} - l_{cort} \leq 5 - l_{cort}$. We can again maximize $\alpha_k^{CO,RT}$ across k (up to 1.1 MHz) for any particular value of l_{cort}. To satisfy the convergence conditions in Theorem 2 and Corollary 1, we need to find l_{cort} such that $\max_k \alpha_k^{CO,RT} \max_k \alpha_k^{RT,CO} < 1 = 0dB$ in the synchronous case and $\max_k \alpha_k^{CO,RT} \max_k \alpha_k^{RT,CO} < 1/\left(\sum_k \gamma(k)\right)^2 \approx \frac{1}{1.66} = -2.2dB$ in the asynchronous case. It turns out that all values of $l_{cort} \in [0, 5]$ km satisfy the convergence conditions as shown in Fig. 3, which means ASB-S2/A2 always converge in the 2-user case for all deployment scenarios.

Similarly we can translate the convergence conditions in the N user case into the constraint on the maximum distance between the CO TX and RT TX. For example, for a network with 5 users, we need to have $l_{cort} < 1225$ m in the synchronous case and $l_{cort} < 1009$ m in the asynchronous case to satisfy the convergence conditions. We want to emphasize that the sufficient conditions for convergence can be loose, and in practice the ASB algorithms always converge.

VII. Simulation Results

In this section, we show the performance of the ASB algorithms, using a realistic simulator based on semi-empirical channel models developed in standards and used extensively in the industry [13]–[15]. We only simulate the performances of the ASB-S1 and ASB-A1 algorithms, which do not involve any high SNR assumptions. These two algorithms always converge in our extensive simulations.

A. Synchronous Transmission Case

Here we summarize a typical numerical example, representative of many experiments we tried, 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 [20] 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 interferences 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 Mbps. User 4 changes its target rate from 0 to 8 Mbps, 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 (i.e., the CO line) in terms of background noise and crosstalk channel gains with

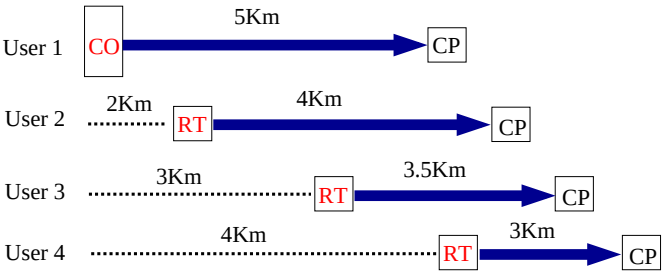


Fig. 4. A four-user mixed CO/RT deployment topology.

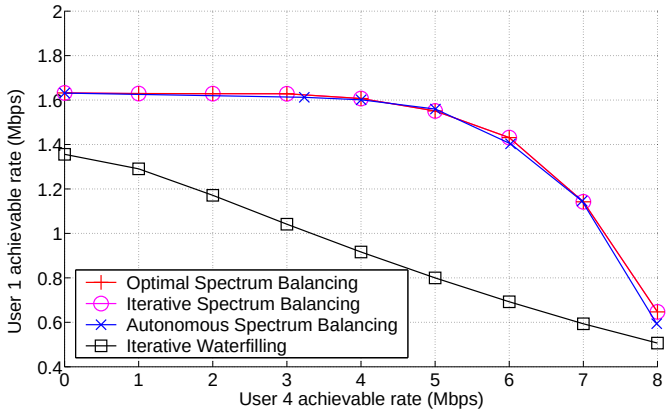


Fig. 5. Rate regions obtained by various DSM algorithms

users in the network. The reference PSD is chosen according to single-user waterfilling without considering the interferences from other users. Based on this reference line definition, we get the rate regions shown in Fig. 5.[5] We can see 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 Mbps on user 1, the rate on user 4 reaches 7.3 Mbps under ASB algorithm, which is a 143% increase compared with the 3 Mbps 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. 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 (using the notations in this paper):

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

where the adjustable part $\frac{w^n(1-\lambda^n)}{\lambda^n}$ is the *same* on all frequencies. User n first adjusts λ^n such that its total power constraint

[5]Note that only ASB uses the reference lien idea.

is tight. If the achieved rate R^n is larger than the target rate $R^{n,\text{target}}$, 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 sub-optimal 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 $R^1 = 1$Mbps, $R^2 = R^3 = 2$Mbps, $R^4 = 3$Mbps for IW and 7.3Mbps for ASB-A1/ISB/OSB.

We also simulate the ASB and IW algorithms in a network with 10 lines, with the line length equal to 5km for the CO line, and 4.5 km, 4.1875 km, ..., 2 km for the RTs. The RTs are located 2, 2.25, ..., 4 km from the CO. The target rate for the CO modem was specified as 1.6 Mbps. 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 Mbps 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 Mbps while still ensuring that the CO modem achieve 1.6 Mbps. ASB algorithm achieves a gain of 122% in RT rate with respect to IW.

B. Asynchronous Transmission Case

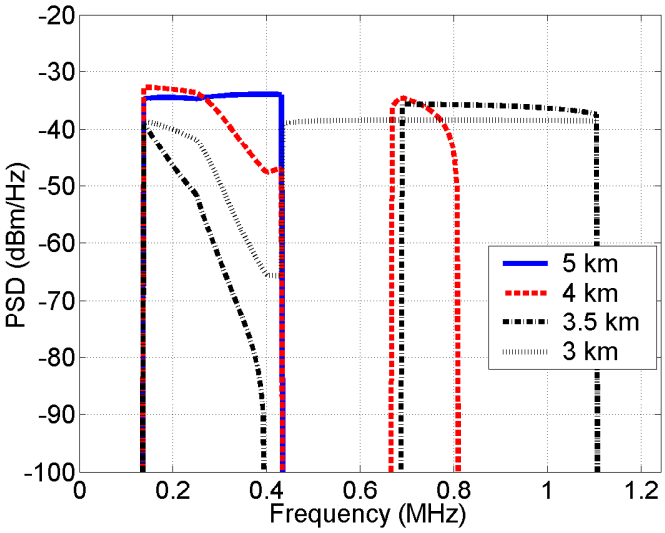
Now consider the case of asynchronous transmission. Here we summarize a typical numerical example comparing the performances of ASB-A1 algorithm with ASB-S1 algorithm. As depicted in Fig. 7, the scenario consists of downstream transmission with two ADSL modems, one 5 km CO line, and one 3 km RT line.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

Figs. 9 and 8 show an example of the PSDs generated by ASB-A1 and ASB-S1. The target rate for the RT is set to 3.85 Mbps. Using ASB-S1, which does not take the effects of the ICI into account when optimizing the transmit spectra, the CO achieves 1.3 Mbps. Using ASB-A1, the CO rate increases to 1.6 Mbps. 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.

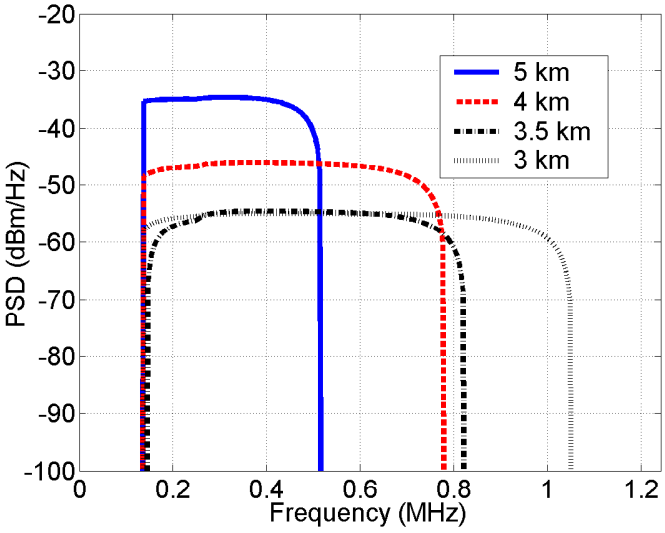
Fig. 10 shows the increase in performance relative to IW achieved by ASB-S1 and ASB-A1 respectively in an asynchronous environment. As we see, even when the modems are not synchronized, ASB-S1 achieves significant gains over IW. Furthermore, if the transmit spectra are further refined through the application of ASB-A1, even further performance gains are possible. For example, if the CO rate is set at 1.4 Mbps, 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

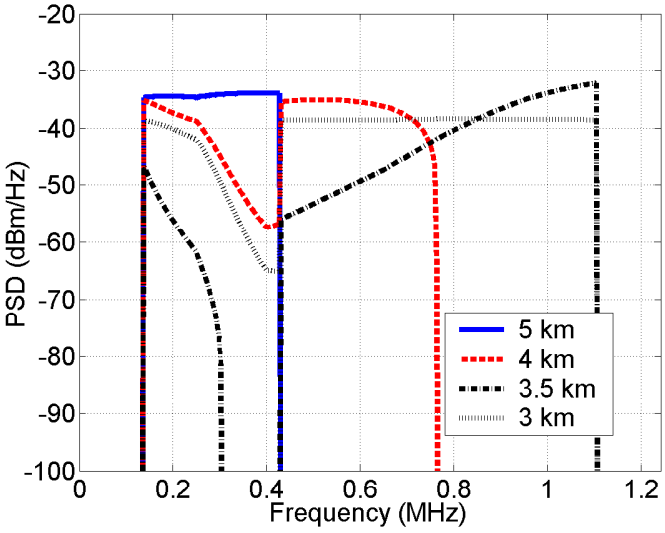
In all previous simulation examples, we choose the reference line to match the longest line in the network. Here



(a) ASB-S1



(b) IW



(c) ISB/OSB

Fig. 6. Transmit spectra with synchronous transmission

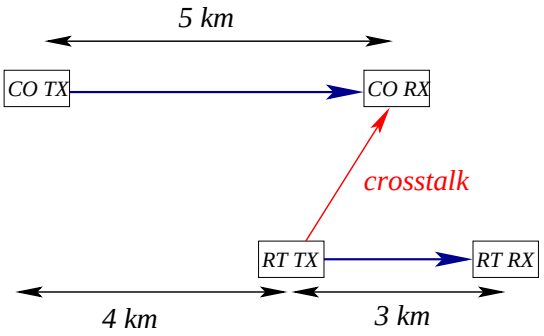


Fig. 7. Mixed CO/RT distribution

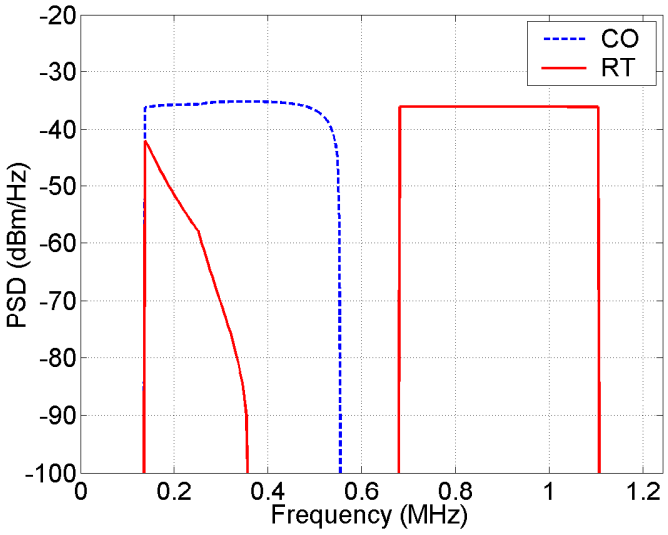


Fig. 8. Transmit spectra with asynchronous transmission: ASB-S1

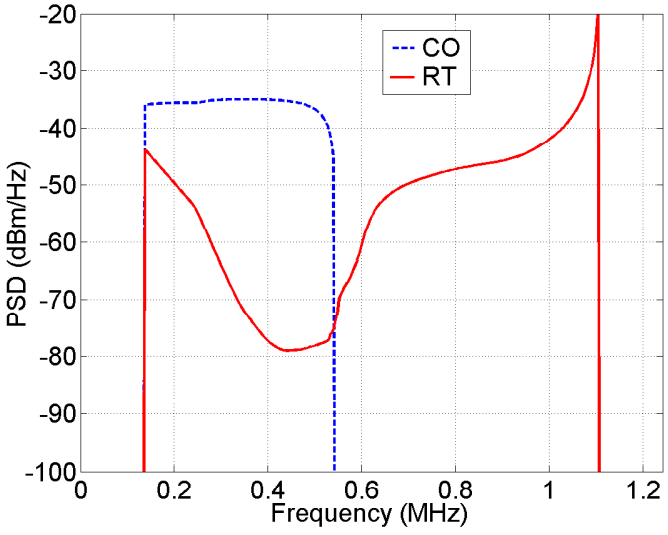


Fig. 9. Transmit spectra with asynchronous transmission: ASB-A1

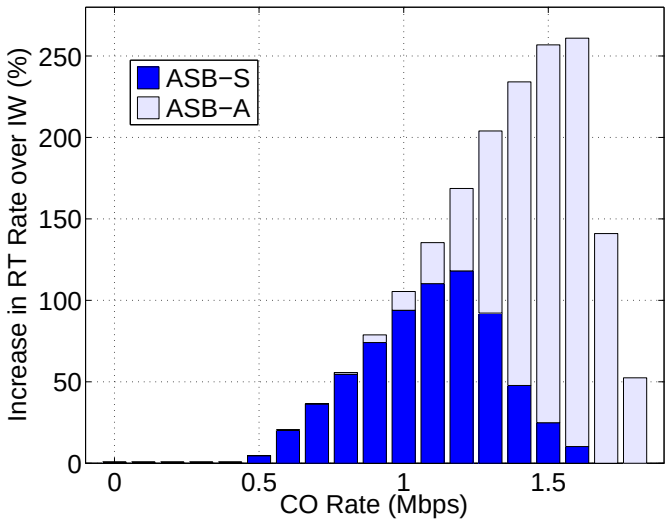


Fig. 10. Performance gains of ASB-S1 and ASB-A1 over IW under asynchronous transmission

we study the sensitivity of the performance to the choice of reference line length. We run simulations in a two user scenario as in Fig. 7, and modems operate synchronously. We vary the reference line length from 4010m up to 6000m to examine the effect on rate region.

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 the 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 RT to the reference line, and the RT is forced to reduce power in the entire frequency band within which the CO transmits. Overall speaking, it is seen that, except in extreme cases, ASB performs well for a broad range of choices of the reference line length.

VIII. Complexity Analysis

Here we compare the complexity of ASB-S1 algorithm with the IW algorithm, which is summarized in Table II.[6] *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

[6]The complexity result of ASB-A1 algorithm is given in Table I, and the corresponding analysis details are omitted due to space limitation.

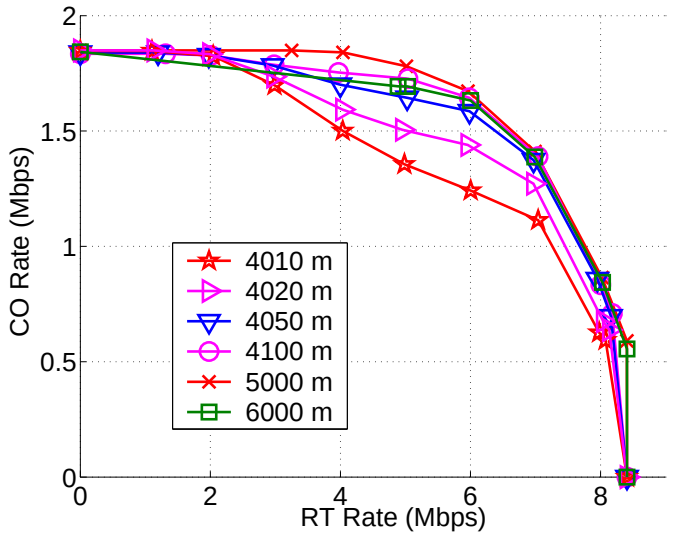


Fig. 11. Sensitivity of ASB-S1 to choice of reference line length

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

necessary for the rates to converge within 1% of the previous cycle.

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 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 $\log_2\left(1/\left(10^{-10}\right)\right) = 34$ 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:[7]

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 [21].
2) Calculate s_k^n based on the optimal water level: K operations.
3) Calculate corresponding integer 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.

[7]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 minimal impact on complexity.

B. Complexity Analysis for ASB-S1

ASB-S1 consists of three levels of iterations, with the outmost cycle iterating through users. Within each cycle, each users runs an outer loop where it updates w_n until the target rate is achieved, and an inner loop where tit 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 [22]. 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.

IX. CONCLUSIONS

This paper presents the Autonomous Spectrum Balancing (ASB) algorithm, the first suite of DSM methods that is simultaneously low complexity, completely autonomous, provably convergent under certain conditions, and achieving close-to-optimal performance in DSL systems. It achieves large performance gain over the state-of-art autonomous algorithm IW, and close-to-optimal performance (established by the centralized OSB algorithm) in a wide range of scenarios. The convergence of ASB is proven for an arbitrary number of users and under channel conditions that are typically satisfied in DSL deployments. 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 in both synchronous and asynchronous transmission case, where the latter is a particularly under-explored research area where only limited, high-complexity heuristics were available.

The key concept that enables ASB to successfully tackle the nonconvex and coupled optimization problem is the reference line, which allows each user to optimize its transmit spectra to achieves its own target rate while minimizing the degradation caused to other users in the frequency-selective interference channel of DSL. ASB applies this approach of "static pricing" coordination in a rigorous manner with provable theoretical properties, leading to significantly enlarged rate region compared with IW. This "reference line" idea can be readily implemented using existing DSM/PBO techniques mandated by DSL standards. Since good choices for reference lines have been made in standards, we can readily apply these in our

algorithms. Although we have focused mainly on ADSL in this paper, ASB is also applicable in VDSL systems and lead to significant performance gains as well.

ACKNOWLEDGMENT

We would like to thank helpful discussions from John Cioffi, Alexander Fraser, Dani Palomar, Chee Wei Tan, and Wei Yu. This work was in part supported by Alcatel-Bell and US NSF Grant CNS-0427677 and CCF-0448012.

REFERENCES

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

[2] R. Cendrillon, G. Ginis, and M. Moonen, "Improved linear crosstalk precompensation for downstream vdsl," in *Proceedings of IEEE International Conference on Acoustics, Speed and Signal Processing (ICASSP)*, 2004, pp. 1053–1056.

[3] G. Ginis and J. Cioffi, "Vectored transmission for digital subscriber line systems," *IEEE Journal on Selected Areas of Communications*, vol. 20, no. 5, pp. 1085–1104, 2002.

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

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

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

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

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

[12] K. Seong, M. Mohseni, and J. M. Cioffi, "Optimal Resource Allocation for OFDMA Downlink Systems," in *IEEE ISIT*, 2006.

[13] "Very-high bit-rate Digital Subscriber Lines (VDSL) Metallic Interface, ANSI Std. T1.424," 2004.

[14] "Very high speed Digital Subscriber Line (VDSL); Functional Requirements, ETSI Std. TS 101 270-1, Rev. V.1.3.1," 2003.

[15] "Very high speed digital subscriber line transceivers 2, ITU Draft Std. G.993.2," 2006.

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

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

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

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

[20] V. Oksman and J. M. Cioffi, "Noise models for vdsl performance verification," ANSI, ANSI-77E7.4/99.438R2, Dec. 1999.

[21] J. Cioffi, "Advanced Digital Communication EE379C Course Reader, Chapter 4 - Multi-channel Modulation. ." [Online] `http://www.stanford.edu/class/ee379c/readerfiles/chap4.pdf`

[22] [Online] `http://www.ping.be/~ping1339/cubic.htm`

[23] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: numerical methods*. Prentice Hall, 1989.

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:

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

where $\beta_k^n = \tilde{\alpha}_k^n \sum_j \frac{\gamma(j-k)}{\tilde{\sigma}_j} \mathbf{1}_{\{\tilde{s}_j > 0\}}$, $n, m \in \{1, 2\}$, $m \neq n$ 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

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

Also define

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

and

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

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 $f^{n,t}\left(\lambda^{n,t+1}\right)=g^{n,t}\left(\lambda^{n,t+1}\right)=0$, which means the PSD converges). From (15) we always have $f^{n,t}\left(\lambda^{n,t+1}\right)=g^{n,t}\left(\lambda^{n,t+1}\right)$. Now we can show that

$$\begin{aligned}
&\max\left\{\sum_k\left[s_k^{1,t+1}-s_k^{1,t}\right]^+,\sum_k\left[s_k^{1,t+1}-s_k^{1,t}\right]^-\right\}\\
&=\max\left\{f^{1,t}\left(\lambda^{1,t+1}\right),g^{1,t}\left(\lambda^{1,t+1}\right)\right\} &(16)\\
&\le\max\left\{f^{1,t}\left(\lambda^{1,t}\right),g^{1,t}\left(\lambda^{1,t}\right)\right\} &(17)\\
&\le\max\left\{\sum_k\left[\sum_j\gamma(k-j)\alpha_j^{1,2}\left(s_j^{2,t}-s_j^{2,t-1}\right)\right]^+,\right.\\
&\quad\left.\sum_k\left[\sum_j\gamma(k-j)\alpha_j^{1,2}\left(s_j^{2,t}-s_j^{2,t-1}\right)\right]^-\right\} &(18)\\
&=\max\left\{\sum_j\left[\sum_k\gamma(j-k)\alpha_k^{1,2}\left(s_k^{2,t}-s_k^{2,t-1}\right)\right]^+,\right.\\
&\quad\left.\sum_j\left[\sum_k\gamma(j-k)\alpha_k^{1,2}\left(s_k^{2,t}-s_k^{2,t-1}\right)\right]^-\right\} &(19)\\
&\le\max\left\{\sum_j\sum_k\gamma(j-k)\alpha_k^{1,2}\left[s_k^{2,t}-s_k^{2,t-1}\right]^+,\right.\\
&\quad\left.\sum_j\sum_k\gamma(j-k)\alpha_k^{1,2}\left[s_k^{2,t}-s_k^{2,t-1}\right]^-\right\} &(20)\\
&=\max\left\{\sum_k\alpha_k^{1,2}\left[s_k^{2,t}-s_k^{2,t-1}\right]^+\sum_j\gamma(j-k),\right.\\
&\quad\left.\sum_k\alpha_k^{1,2}\left[s_k^{2,t}-s_k^{2,t-1}\right]^-\sum_j\gamma(j-k)\right\} &(21)\\
&\le\left(\sum_j\gamma(j)\right)\max_k\left\{\alpha_k^{1,2}\right\}\\
&\quad\cdot\max\left\{\sum_k\left[s_k^{2,t}-s_k^{2,t-1}\right]^+,\sum_k\left[s_k^{2,t}-s_k^{2,t-1}\right]^-\right\} &(22)\\
&\le\left(\sum_k\gamma(k)\right)^2\max_k\left\{\alpha_k^{1,2}\right\}\max_k\left\{\alpha_k^{2,1}\right\}\\
&\quad\cdot\max\left\{\sum_k\left[s_k^{1,t}-s_k^{1,t-1}\right]^+,\sum_k\left[s_k^{1,t}-s_k^{1,t-1}\right]^-\right\} &(23)\\
&<\max\left\{\sum_k\left[s_k^{1,t}-s_k^{1,t-1}\right]^+,\sum_k\left[s_k^{1,t}-s_k^{1,t-1}\right]^-\right\}, &(24)
\end{aligned}$$

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 $\sum_k[x_ky_k]^+\le\sum_k x_k[y_k]^+$ for all $x_k\ge0$ and y_k, (21) follows by exchanging the summation order of k and j, (22) follows by using the circulant property of γ (23), i.e., $\sum_j\gamma(j-k)=\sum_j\gamma(j)$, (23) by applying the arguments from (16) to (22) again, and finally (24) follows by the condition in Theorem 2. This shows that the ASB-A2 algorithm is a contraction mapping form an initial PSD values, thus globally converges to a unique fixed point [23, Page 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

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

The rest of the proof can be obtained similar as in Theorem 2 with the following:

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

For the sequential update case, the convergence can be proved by combining Lemma 1 and proof of Theorem 3.4.1 in [10]. First, define $D_{\boldsymbol{s}^t, \boldsymbol{s}^{t'}}(n) = \max \left\{ \sum_k \left[s_k^{n,t} - s_k^{n,t'} \right]^+ , \sum_k \left[s_k^{n,t} - s_k^{n,t'} \right]^- \right\}$, and $\boldsymbol{D}_{\boldsymbol{s}^t, \boldsymbol{s}^{t'}} = \left\{ D_{\boldsymbol{s}^t, \boldsymbol{s}^{t'}}(n), \forall n \right\}$. 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. ■