Contents

Chapter 15

The Gaussian Interference Channel

The Gaussian Interference Channel, **IC**, is a set of linearly inter-coupled additive Gaussian Noise channels with intersymbol interference possible on all (including crosstalking) paths. No coordination in modulation and coding of transmit signals nor in reception and detection of receive signals is allowed, although the existence of a master design for all users is allowed in the **IC**. Because best design of signals (from Chapter 12) for the **IC** is that all users are Gaussian, then the master design essentially becomes a coordinated assignment of power spectral densities[1] of each of the otherwise uncoordinated users on the **IC**. Such central assignment of spectra is here designated as a "Centrally Controlled IC" or CCIC. When no master design of spectra is allowed, this text calls the situation "Distributed Control IC" (DCIC).

Section 15.1 provides models and basic fundamental capacity regions for both the CCIC and the DCIC. Each of the **IC** users' channels may themselves be vectored. The results of Section 15.1 will apply to such channels as well as to any set of linear inter-coupled additive Gaussian noise channels. Some of the vector users may themselves be sets of users that could have multiple-access or broadcast constraints within the set, and the results of Section 15.1 will also apply to that group as a sum-rate user (that either can be entirely, or cannot be at all, detected as per Chapter 12's definition of a user). Generation of **IC** capacity regions relies heavily on the use of successive decoding in receivers, where it is possible to decode other Gaussian users first before decoding the user of interest. Such a restriction implies that all the decoders know the codes and spectra of the other users and that all such spectra have been designed centrally, a constraint that may not be feasible for practical or regulatory reasons. Section 15.2 then proceeds to the best optimization when no central control is possible and the receivers use no successive decoding, namely "iterative water-filling" for the **IC**. Two forms of IW are developed with the rate-adaptive form[2] not surprisingly leading to a situation that is near worst-case in a number of situations, while the "fixed-margin" form creates a level of politeness that allows the DCIC to perform significantly better than the rate-adaptive or margin-adaptive cases. Some examples of large gains are presented in Section 15.2 along with a discussion of Brady's worst-case spectrum design for an IC. Such a worst-case differs from that of broadcast channel and there shown to be often not far from the rate-adaptive IW solution.

The definition of a DCIC raises the question of "Just how much central control is really feasible?" Both practical and regulatory restrictions may impose limits on central control. The (fixed-margin) IW of Section 15.2 essentially presumes that only the data rates of each of the users are centrally imposed along with a "minimize power" constraint at that rate. Such a central data-rate assignment is a minimum central control for any multi-user network strategy. Without central rate assignment either at design time, by mutual covenant or standardization, or by a regulatory agency, the multi-user problem essentially degenerates into a set of independent single-user problems (whose joint solution may not be good for the set of users). Then dimensions are simply assigned in mutually exclusive sets to each user for use, which is often not near a best strategy. With central rate assignment, the fixed-margin IW is appropriate when receivers do not use successive decoding. The presumption that no users' receivers employ successive decoding is a practical constraint that reflects the other users' Gaussian codes, which

[1] More generally speaking one can read spectral densities as autocorrelation matrices.

[2] or equivalently margin adaptive, but not fixed margin.

prevent another receiver from adaptively determining the signal without à priori knowledge of the code used.

Section 15.3 raises the possibility that central control might impose spectra for all users but no successive decoding is allowed for feasibility reasons. Such a problem is of practical interest although equal to neither an CCIC (which would allow successive decoding) nor a DCIC (which would not allow the central design of spectra). Cendrillon's approach to this problem provides some interesting centralized "bit-swapping" methods known as "optimum spectrum management" or "optimum spectrum balancing." The methods are optimum under a set of constraints of central spectrum design, presumption of Gaussian signals, and that no receivers may use successive decoding. These methods will also presume a synchronization of all channels so that vector DMT signals may be aligned on all channels, an assumption that also tends towards the more coordinated side of the CCIC than the DCIC. Those constraints and the method are reviewed in Section 15.3. Some improvements are illustrated with respect to IW that accrue to the improved central ability to allocate spectra also in Section 15.3. Some approaches to simplification of the optimization algorithm, which may be too complex when the set of conditional optimalities is fully observed and applied, also appear in Section 15.3.

Section 15.4 introduces the general area of "multi-level water-filling" as a somewhat less centralized approach that imposes some power spectral density guidance centrally, but otherwise returns to the use of Section 2's IW independently for each of a set of frequency bands. In many cases, this approach leads to the same results for considerably less complexity and retains some level of local adaptability for each user to react to changes within its own channel.

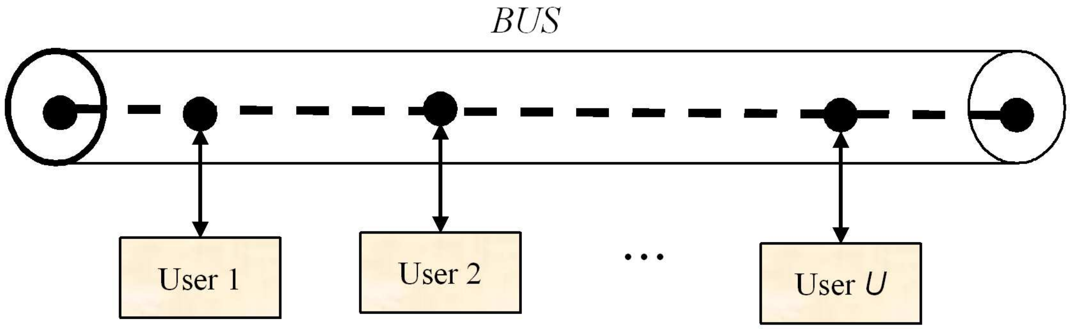

Figure 15.1: Bus architecture for IC.

15.1 Gaussian Interference Channel Fundamentals

Again, this chapter deals only with additive Gaussian channels with linear distortion and linear crosstalk, as in Chapters 13 and 14.

15.1.1 The Vector Gaussian IC (IC) channel model

The **IC** is mathematically described by

$$\boldsymbol{y} = H \boldsymbol{x} + \boldsymbol{n} \tag{15.1}$$

$$\underbrace{\begin{bmatrix} \boldsymbol{y}_U \\ \vdots \\ \boldsymbol{y}_1 \end{bmatrix}}_{L_y NU \times 1} = \underbrace{\begin{bmatrix} H_{UU} & \dots & H_{U1} \\ \vdots & \ddots & \vdots \\ H_{1U} & \dots & H_{11} \end{bmatrix}}_{L_y NU \times L_x (N+\nu) U} \underbrace{\begin{bmatrix} \boldsymbol{x}_U \\ \vdots \\ \boldsymbol{x}_1 \end{bmatrix}}_{L_x(N+\nu)U \times 1} + \underbrace{\begin{bmatrix} \boldsymbol{n}_U \\ \vdots \\ \boldsymbol{n}_1 \end{bmatrix}}_{L_y NU \times 1} . \tag{15.2}$$

Each entry of the H matrix can be $L_y N \times L_x(N+\nu)$, meaning that each user may have L_x transmitters and symbols of length $N+\nu$ dimensions while transmitting to L_y receivers. Each receiver ignores any guard periods. The overall H matrix is thus $L_y NU \times L_x(N+\nu)U$.

Such an **IC** can be achieved by a few architectures that initially may appear different. Figure 15.1 illustrates the bus architecture where several users attach to a common medium. The medium may be a single wire (some old earliest forms of "Ethernet" used a single coaxial cable with "T" connectors), a wireless medium, or a set of several (say 8 for a byte) wires for a computer bus. There may or may not be a central controller that arbitrates connection to the bus. For instance, in a computer processing environment, the bus is some multiple of 8 wires and is controlled by a host processor, creating a CCIC. In another example, the medium may be a unregulated wireless band where any of the users may attempt to access at any time. The old ethernet methods, known as the original IEEE 802.3 ethernet standard, and the some wireless local-area network methods, particularly those known as IEEE 802.11 a,b,g, or n use a method called Carrier-Sense Multiple Access (CSMA) to avoid the absolute need for a central controller.

EXAMPLE 15.1.1 (CSMA) Carrier Sense Multiple Access (CSMA) is actually a strategy for both interference and multiple-access channels even though the words "multiple access" appear in the well-known acronym. CSMA attempts a somewhat statistical construction of orthogonal multiplexing by allowing all users to transmit whenever they like. If the users don't transmit too often, it is unlikely they will collide or crosstalk. So most of the time on a lightly used network, CSMA implements orthogonal time-division multiplexing by random infrequent time use. In effect, CSMA is TDMA when the channel is lightly loaded. However, there is a probability that two users will transmit at the same time, and then crosstalk occurs. As use increases, so does this probability. Readers of this text may see this "collision" as an opportunity for multi-user methods in at least the receiver. Early CSMA designers saw this crosstalk event as a negative, and thus the change of the name "crosstalk" to the more ominous sounding "collision."

Each CSMA receiver attempts to decode all messages (eventually discarding any not addressed to it), and an error detection code is checked by the receiver (CRC codes of Chapter 10 are typically used). If the error-detection finds no errors, an acknowledgment is sent back to the originating message location. If the transmit's co-located receiver successfully decodes this acknowledgment, the original message is assumed to have been successfully received. If after a period of time, no such acknowledgment is received, the originating transmitter attempts again. To avoid repeated collisions, the retransmission is attempted after a random period of time. Realizing that the delay time of a network to respond is typically much less than the length of a packet message transmitted, "collision detection" often augments CSMA to form CSMA-CD. Any receiver on the network attempts to estimate if two or more signals are simultaneously present and if so immediately emits a "jamming" signal that informs all users' receivers to then silence their local transmitters (if transmitting presently or recently) and to retransmit after a random period. In this way essentially the transmitters do not "waste time" continuing in a collision – such a strategy again presumes a collision is bad, and no multi-user detection is present other than the collision detector itself. CSMA, while clearly not efficient when users are often active, is one of the most common multiplexing methods used in computer networks today, be it wire-line or wireless.

For wire-line , the most common"carrier sensing" modulation is typically implemented with one of two types of transmission, wireless' BPSK ($\bar{b} = .5$ QAM) and/or Ethernet's Manchester ($\bar{b} = .5$ PAM on 2 dimensions with one wasted – See Chapter 1). Both of these transmission methods waste dimensions but guarantee a carrier transition in each used bit period. Thus, the collision or presence of a signal is very easy to detect because of the guaranteed transition. This is hopelessly inefficient transmission in a network that has high SNR – not to mention the long delays holding a signal before it is acknowledged tend to reduce average throughput with finite memory buffers. Nonetheless, CSMA led to easy implementations in early networks at the expense of a great deal of efficiency. 10Base-T transmission for instance still uses Manchester encoding because of the old Ethernet legacy even though the carrier is no longer necessary in the hub architecture (100Bbase-T and 1GBase-T improved upon this and do not use Manchester encoding nor CSMA). Use of CSMA leads to similar efficiencies in wireless, but nonetheless is often found because of its simplicity. Most wireless local-area-networks ("wifi") use CSMA.

Outside of the carrier-sensing (or "collision detection," the CSMA approach of Example 15.1.1 essentially allocates dimensions (in this case, time dimensions) to users so that they do not interfere with one another. Such "orthogonal multiplexing" was discussed in Chapter 12 and is often not the best use of a limited channel's resources. Nonetheless, a vector model such as in (15.2) could be found for such a channel by individually determining the crosstalking transfer functions from each user to every other user. The noise for each user would be measured at its receiver. The configuration in Figure 15.1 suggests that each user of the bus IC may independently transmit or receive messages. Such a system may be best modeled by a maximum of $U(U-1)$ "users" with each user possible desiring to send messages to every one of the others but not to itself. WiFi systems with CSMA may have an "access point" to which all transmissions are sent and from which all receptions emanate. However, the channel still uses the CSMA protocol and is better modeled as an **IC** than as a **MAC** uplink combined with a **BC** downlink. In WiFi, the access point receiver simply allows higher-level access to other networks and an ability to forward messages to other users[3], but usually does not practice coordination at the modulation level.

By contrast to the bus, the "binder" model of Figure 15.2 is U transmitters and U receivers each trying to communicate only with its counterpart, but experiencing interference from the other users

[3]Some IEEE802.11 (n) systems may use advanced cancelation methods with multiple antennas and are better modeled as uplink vector multiple access and downlink vector broadcast where simultaneous transmissions of multiple users may be allowed and separated from one another via the methods of Chapters 13 and 14.

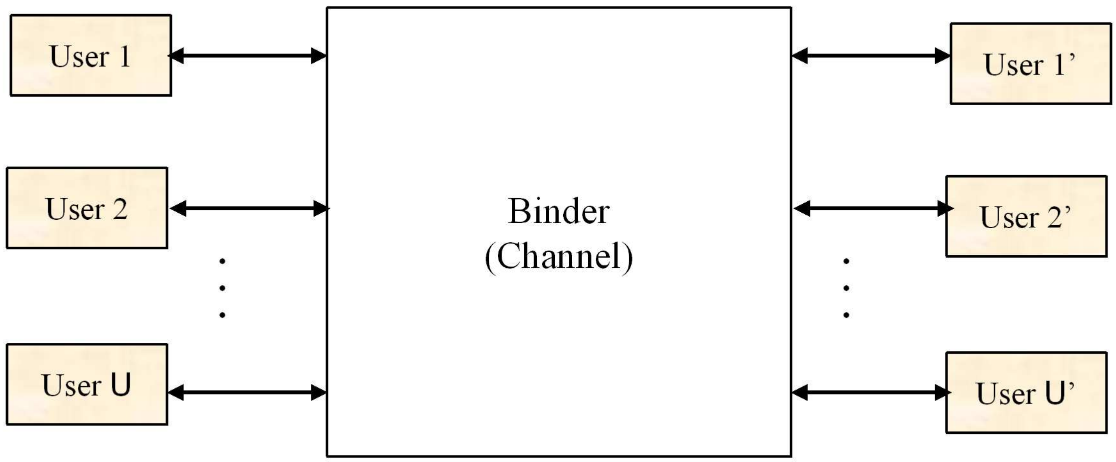

Figure 15.2: Binder architecture for IC.

of the channel. There may or may not be central control in the binder. While the architecture is different from the bus, it is also an example of the **IC**. The binder system may be "duplex" in that both directions of transmission are desired, in which case it may be best modeled as having $2U$ users. The binder situation corresponds to DSL systems where co-location of different service providers (who will not cooperate through a central controller or "vector" DSLAM of Chapters 13 and 14) and thus model a DCIC, albeit possibly with some standardization on frequency use of the users. The binder model also applies to a single DSLAM or to an "access point" in WiFi if the co-located transmitters and receivers do not cooperate and simply just otherwise occupy the same piece of equipment without knowledge of other users' choices of spectra and coding.

The binder and bus models fundamentally both reduce to the vector channel model of (15.2) with only the possible number of users changing, but the basic structure of a known transfer function from every possible transmitter of a message to every possible receiver of that message still holds. This text conceptually uses the binder model. However, with the developments of this section, it is always possible to cast any **IC** into the format of (15.2) and then subsequently view it as a binder channel with an appropriate number of users.

User u's transmit spectra for a finite symbol size is generally replaced by $R_{\boldsymbol{xx}}(u)$, which as $N \rightarrow \infty$ could be modeled as a Gaussian power spectral density for stationary systems. The use of the autocorrelation $R_{\boldsymbol{xx}}(u)$ is more general and allows for finite-length packets and possibly modeling of time variation. There will be an autocorrelation matrix for each user. These aucorrelation matrices will be known to all users in the case of the CCIC, but will not be known (nor will the entries of off-diagonal elements of H in the DCIC). A subset of users may be grouped together in an IC, in which case the user index becomes a vector $\boldsymbol{u}$. From the viewpoint of the IC, the users $u \in \boldsymbol{u}$ are all the same user with the rate being the sum of subset users' data rates.

15.1.2 The Centrally Controlled Gaussian IC (CCIC)

Figure 15.3 depicts the CCIC. Central assignment of bits per user (per dimension) and corresponding $R_{\boldsymbol{xx}}(u)$ is possible with the CCIC through the central controller that knows all noise autocorrelation matrices $R_{\boldsymbol{nn}}(u)$ and channel H_{ij} entries. This CCIC is the traditional IC of other texts.

The capacity region

The capacity region of the IC follows from the general IC capacity region in Chapter 12 with some simplification in the Gaussian case. Each receiver is allowed to use a GDFE receiver for any possible order of users that receiver sees appropriate.

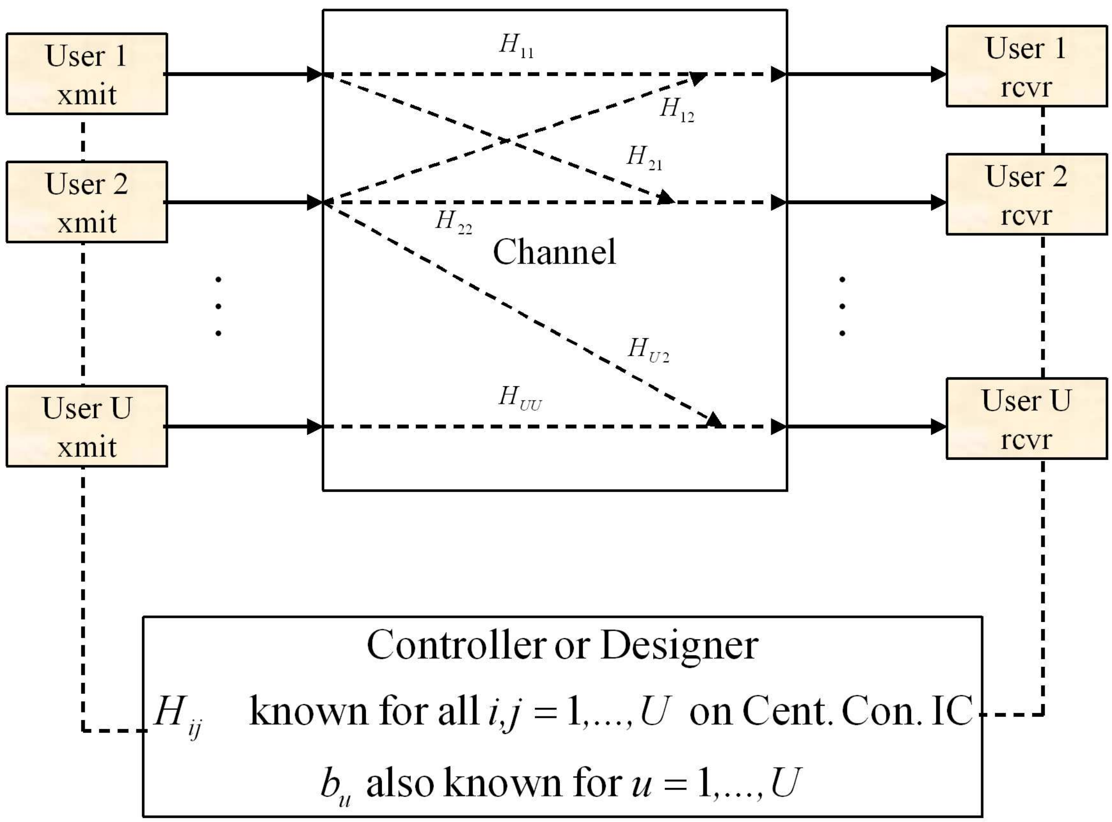

Figure 15.3: Centrally Controlled **IC**.

A first step is to introduce an order vector

$$\boldsymbol{\pi} = [\boldsymbol{\pi}_U, \ ..., \ \boldsymbol{\pi}_1] \quad , \tag{15.3}$$

where $\boldsymbol{\pi}_u$ is the order selected by receiver u. Each user's receiver may select any of $U!$ orders. There are thus

$$|\boldsymbol{\pi}| = (U!)^U \tag{15.4}$$

possible orders when vieweed over the set of users. For any given order vector $\boldsymbol{\pi}$ and given input autocorrelation matrix $R_{\boldsymbol{xx}}$ (viewed as a block diagnonal matrix with each of the user autocorrelation matrices as one of the blocks), receiver u may use a GDFE for its component order π_u to achieve a vector of user bits/symbol denoted $\boldsymbol{b}_u(\boldsymbol{\pi}, R_{\boldsymbol{xx}})$. The particular order will determine the data rates for each of the users with all later users in the order at receiver u viewed as noise. A Gaussian code for each user could be designed to achieve this corresponding data rate to receiver u. The notation $\boldsymbol{b}_i(u, \boldsymbol{\pi}, R_{\boldsymbol{xx}})$ denotes the u^{th} component (user u data rate) at receiver i.

The data rates achievable for this order and input autocorrelation may not be achievable at other receivers $i \neq u$. Equivalently, user u's data rate at receiver u may not be achievable at some or all of the other receivers. Indeed, there will be a minimum data rate for user u across all receivers for the given order vector $\boldsymbol{\pi}$ and the given input autocorrelation matrix $R_{\boldsymbol{xx}}$:

$$b_u(\boldsymbol{\pi}, R_{\boldsymbol{xx}}) = \min_i \boldsymbol{b}_i(u, \boldsymbol{\pi}, R_{\boldsymbol{xx}}) \quad . \tag{15.5}$$

This minimum can be achieved at all receivers for the given order and input autocorrelation. A vector of these minimum rates can be constructed as

$$\boldsymbol{b}(\boldsymbol{\pi}, R_{\boldsymbol{xx}}) = \bigotimes_{u=1}^{U} b_u(\boldsymbol{\pi}, R_{\boldsymbol{xx}}) \quad , \tag{15.6}$$

where $\bigotimes$ corresponds to Cartesian product (or simply form an ordered U-tuple). Each element data rate of $\boldsymbol{b}(\boldsymbol{\pi}, R_{\boldsymbol{xx}})$ can be achieved at all receivers and any user data rate that exceeds its corresponding entry in $\boldsymbol{b}(\boldsymbol{\pi}, R_{\boldsymbol{xx}})$ cannot be achieved for this order $\boldsymbol{\pi}$ and this input $R_{\boldsymbol{xx}}$ at one or more receivers (and thus a higher data rate would be a single-user GDFE capacity violation at one or more receivers for decoding). There are many $((U!)^U 0$ orders and so there are many points $\boldsymbol{b}(\boldsymbol{\pi}, R_{\boldsymbol{xx}})$. Any time-sharing of the designs corresponding to these points is allowed; equivalently the convex hull of the region formed by the set of points over all orders for any given $R_{\boldsymbol{xx}}$ is achievable:

$$A(\boldsymbol{b}, R_{\boldsymbol{xx}}) = \bigcup_{\boldsymbol{\pi}}^{conv} \boldsymbol{b}(\boldsymbol{\pi}, R_{\boldsymbol{xx}}) \quad . \tag{15.7}$$

Any point outside this convex hull has at least one data rate that for the given $R_{\boldsymbol{xx}}$ has at least one receiver that cannot decode at least one user that it must decode no matter what order is used. Such a point then violates a single-user capacity limit for all orders and the given input $R_{\boldsymbol{xx}}$. Finally then,

$$c_{IC}(\boldsymbol{b}) = \bigcup_{R_{\boldsymbol{xx}}}^{conv} A(\boldsymbol{b}, R_{\boldsymbol{xx}}) \tag{15.8}$$

where the convex hull over all possible input spectra allows autocorrelation matrices for each independent user that each must satisfy the particular user's energy constraint

$$\text{trace}\{R_{\boldsymbol{xx}}(u)\} \leq \mathcal{E}_u \quad . \tag{15.9}$$

15.1.3 The Distributed Control Gaussian IC (DCIC)

The distributed control aspect of the DCIC is not truly an extra constraint for the **IC** because a designer could presumably guess the correct codes (and receivers if given sufficiently long time could presumably ascertain the constellations and codes used on all other users if those codes were not quite, but almost,

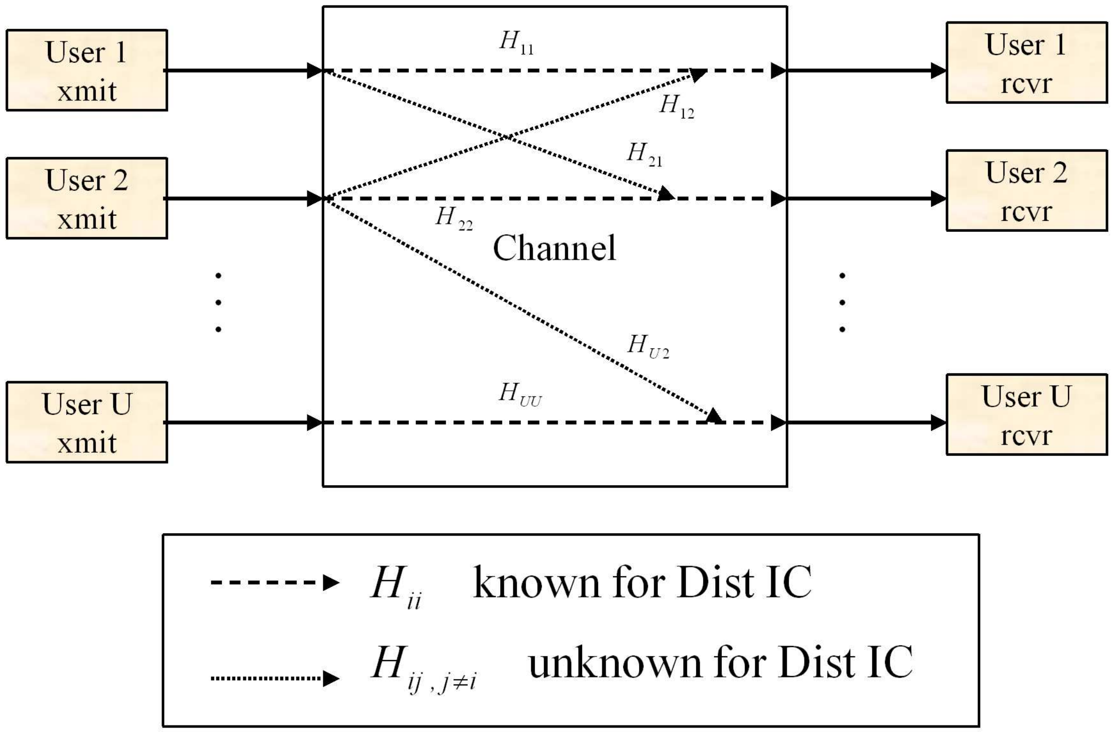

Figure 15.4: Distributed Control **IC**.

Gaussian). Or, the receivers could "guess" right on code designs also. However, the intent of the definition of the DCIC is to refine the constraints in developing a coding strategy for the IC where the central design of spectra and bit distributions may not be feasible.

Instead, the designer for the DCIC may presume only that the following 4 items are known as suggested in Figure 15.4:

1. the received noise autocorrelation $R_{noise}(u)$ (presumed Gaussian including contributions in aggregate from all other users and the Gaussian noise)

2. its own channel H_{uu} (but not any H_{iu} where $i \neq u$)

3. The transmit autocorrelation $R_{\boldsymbol{xx}}(u)$ of its own channel

4. the bit distribution and thus total rate of its own channel $b_{u,n}$ and b_u respectively.

Furthermore, successive decoding will be presumed infeasible with the DCIC. The designer can assume that the data rate for each user can be fixed centrally.

Section 15.2 will address this problem with fixed-margin IW, which often converges for such a channel, but is not guaranteed to converge in all cases.

15.2 Iterative Water-filling

Iterative water-filling (IW) is the same basic procedure as originally defined in Chapter 13 for the **MAC**. However, IW is not necessarily convergent on the **IC**, but has been observed in practice to converge in a wide range of situations. IW has been proven to converge in various situations of interest by Luo and Pang[4]. One situation is where the channel is symmetric (meaning that the transfer function of interference from user i to user j is equal in magnitude to the transfer function from user j to i for all i and j, and the noises are also symmetric otherwise. Perhaps of greater interest (unlikely the noises are symmetric) is that of diagonally dominant channels. Such channels imply that $|H_{uu}(f)| >> |H_{ui}(f)|$ when $i \neq u$. In effect the transfer function of the user's channel is significantly larger than the transfer function from any other user into this same user. Downstream DSL satisfies this constraint if all signals are launched from a common location and usually even if they are transmitted from different points. IW has been observed to converge in all known situations. There may be many points to which IW can coverage on the IC (but any may be acceptable improvements over some less adaptive design approaches), unlike the unique convergence point of the **MAC**. IW is best used when all receivers treat all other signals as noise, and thus best applicable to the DCIC. Essentially, each user being as polite as possible is about the only acceptable solution given a situation where no successive decoding is used by any of the IC receivers. As the constraints on central control are relaxed (but the presumption of no successive decoding retained), additional improvements are possible as in Sections 15.3 and 15.4.

15.2.1 The IW Algorithm for the DCIC

In IW for the DCIC, each user water-fills by treating all other crosstalking signals on the line as Gaussian noise. The spectrum of user u water-fills using the curve corresponding to the ratio of that total noise normalized to the known channel gain $|H_{uu}|^2$, with energy specifically given on tone n as

$$\lambda_u = \mathcal{E}_{u,n} + \frac{\Gamma \sigma_n^2 + \sum_{i \neq u} |H_{ui}|^2 \cdot \mathcal{E}_{i,n}}{|H_{uu,n}|^2} \quad . \tag{15.10}$$

In actual use, each user would presumably implement a water-fill-based loading algorithm that treats all other users as noise, typically with bit-swapping as a reasonable approximation to water-filling. All users may be swapping simultaneously. For simulations and evaluation, it is usually more convenient to hold all other users constant and implement a water-fill loading algorithm for the user of interest. This process iterates through all users several times until the spectra of all users have converged.

Figure 15.5 illustrates the off-line emulation algorithm for iterative water-filling. Basically each user successively water fills as if all others are noises. After a few to several passes through the procedure (as indicated by j_{max} - typically $j_{max} = 5$ is sufficient), it converges to stable spectra for all users.

The use of FM water-filling is important because it corresponds to polite use of power where no user has excessive margin. In particular, the non-unique convergence point to which IW converges is usually then a good one. Section 15.2.3 assesses the impact of using instead rate-adaptive (or equivalently margin adaptive) water-filling algorithms.

Distributed Implementation of IW and related algorithms

Essentially IW can conform to a distributed implementation where each modem uses a water-filling algorithm, treating all other users as Gaussian noise added to whatever Gaussian noise is already present on the channel. Thus, each receiver independently implements a water-filling algorithm without knowledge other than other users are also presumed to be water-filling in a general sense. FM water-filling where each user minimizes their own energy use in achieving a certain data rate and margin is preferred as stated above.

[4] Eurasip Journal on Applied Signal Processing, 2006.

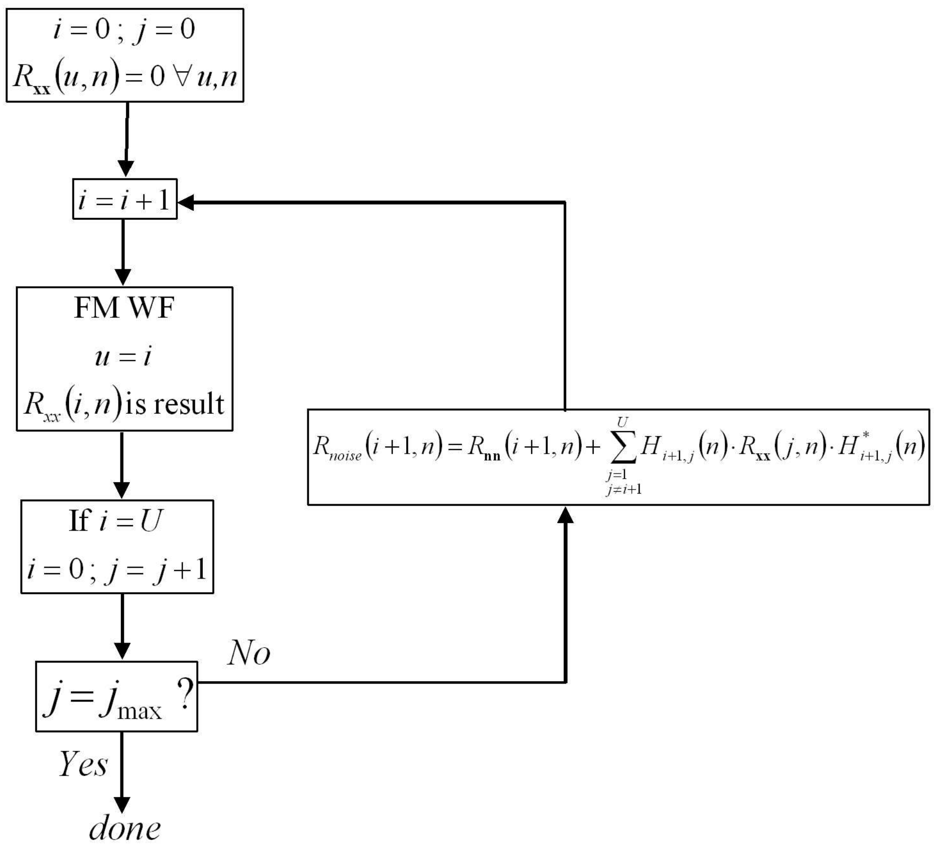

Figure 15.5: Iterative Water-filling (fixed margin).

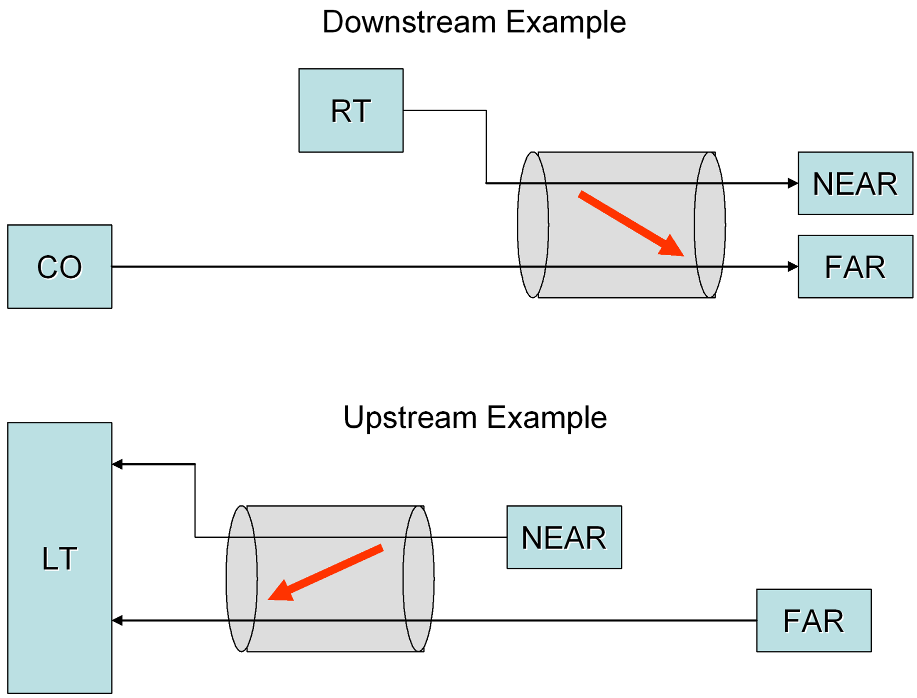

Figure 15.6: Illustration of Near/Far IC problem.

15.2.2 Some DSL Examples

Figure 15.6 illustrates DSL's equivalent of the wireless "near/far" problem where signals from a near transmitter overwhelm signals from a far transmitter at a receiver location. This situation pervades emerging DSL systems where fiber-fed systems move transmitters closer to users, but existing users cannot (or are not) moved to the fiber system[5]. Crosstalk in DSL channels increases with the square of frequency, attenuates like the channel, and has a strength proportional to the common length of the neighboring telephone lines. Details of models are left to standards and other textbooks, but the examples here all use crosstalk transfer functions well-known to be representative in North America. In the near/far situation, the fiber-fed terminal's transmitter couples as "Far-End Crosstalk (FEXT)" into the long line's receiver. The reduction in data rate can be very high on the long line, typically a reduction of a factor of 4 to 5 if MA after the near loop energizes and starts creating crosstalk. The far user then loses performance.

An early example appears in Figure 15.7. The far loop was originally happily working at 1.5 Mbps service level. An remote fiber-fed DSL terminal was placed at 10,000 feet from the central office and also transmitted data using MA water-filling at 1.5 Mbps. The data rate on the long loop then dropped to 300 kbps, a factor of 5 loss in data rate. A third user from the remote terminal also transmits on a longer loop at 1.5 Mbps using MA water-filling. Figure 15.7 plots the IW rate region (each dot is a pair of data rates on the two long lines that was achieved with all lines using FM water-filling and the short line using 1.5 Mbps with 6 dB of margin). The data rate 1.5 Mbps is possible (6 dB of margin was also used on the two long lines) so the original data rate is restored, and all 3 customers operate at the data rate they purchased. Thus, while IW may not be optimum, it very much solves an otherwise catastrophic problem. The key was the short-user using FM water-filling and thus being polite by fixing its margin to some acceptably low value. In some cases, a known long line may need to use margin adaptive (full power) to ensure convergence in this situation of IW to a good point.

[5]The reason for not moving the customer is that they lose service during the mode, which can be up to a week, and it is extra cost to the telephone company.

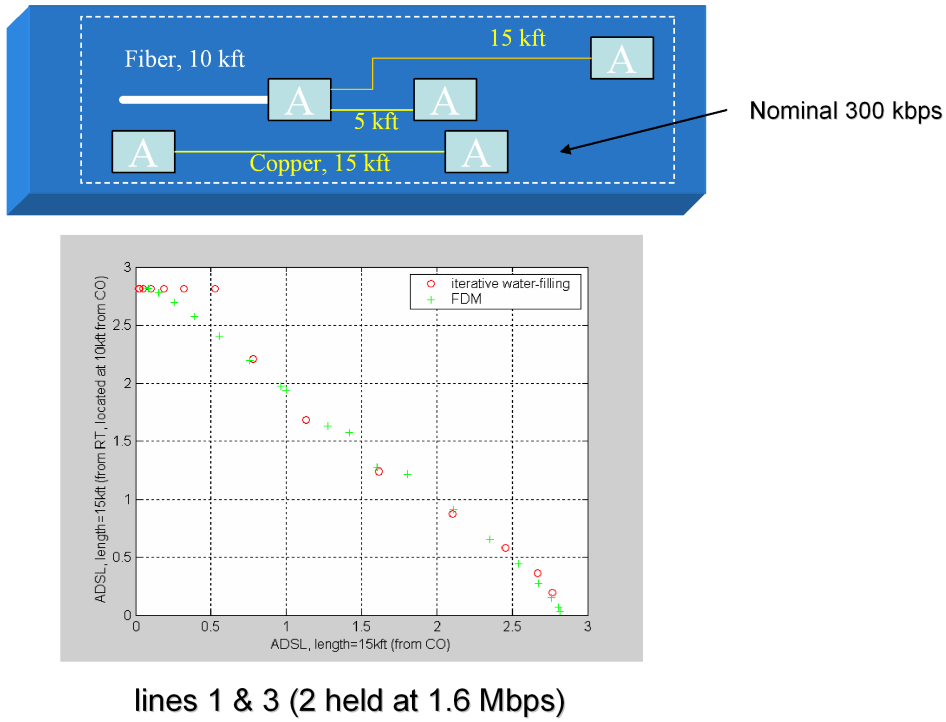

Figure 15.7: Mixed Binder DSL Example.

Figure 15.8 illustrates a situation where the loops (25 of them) all have high crosstalk into one another, but all are the same length (so no near far). In this simulation, the number of users was actually 50 because the lines are all used bi-directionally. Echo cancelation is used on any loop (see Chapter 4) to isolate upstream and downstream transmissions, but crosstalk into all other loops is presumed. The lower curve is a 64PAM system in common use for symmetric transmission and known as SHDSL. The plot shown for SHDSL very much represents its best performance in the field. For the upper curve a VDSL system with potential use of up to 4096 4.3125 kHz tones is used with no restriction on frequencies used up or down. All lines use FM IW and the bandwidth is determined adaptively for each. As is clear, the IW doubles to triples the rate or essentially adds a mile of range at any given data rate. Such an improvement is solely caused by the better adaptive polite determination of spectrum use. The symmetric transmission scheme would need to ensure that all lines use FM water-filling at the same data rate to see the gains shown, and such common understanding might have to be imposed by standards if unbundling (multiple service providers each using lines in the binder) is present. Nonetheless, all customers (and service providers) would gain substantially.

Figure 15.9 is the converged spectra of IW for one of the lines, both upstream and downstream. The lower frequencies are used in both directions since crosstalk is low at lower frequencies. As the frequencies increase, the IW produces a frequency-division-like separation of up and down transmissions because of the stronger crosstalk at those frequencies.

15.2.3 Brady's Worst-Case Noise

While the **IC** capacity region is an outer bound for the performance, and therefore on the DCIC, an interesting question is a worst-case bound, namely one in which the users are hostile to one another and mutually cost largest degradation. Aversion of such a worst-case would then be prudent in design of a DCIC if possible.

M. Brady has developed an approach to determining the worst-case noise. Such a worst-case problem is somewhat ill-posed in that all users could simply be obnoxiously impolite transmitting as much power

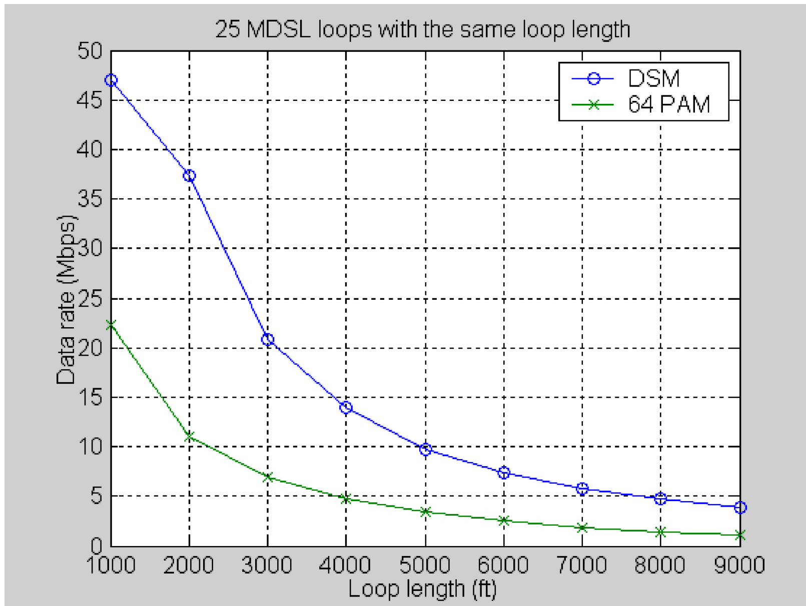

Figure 15.8: Improvement of IW over current static design in symmetric DSL transmission.

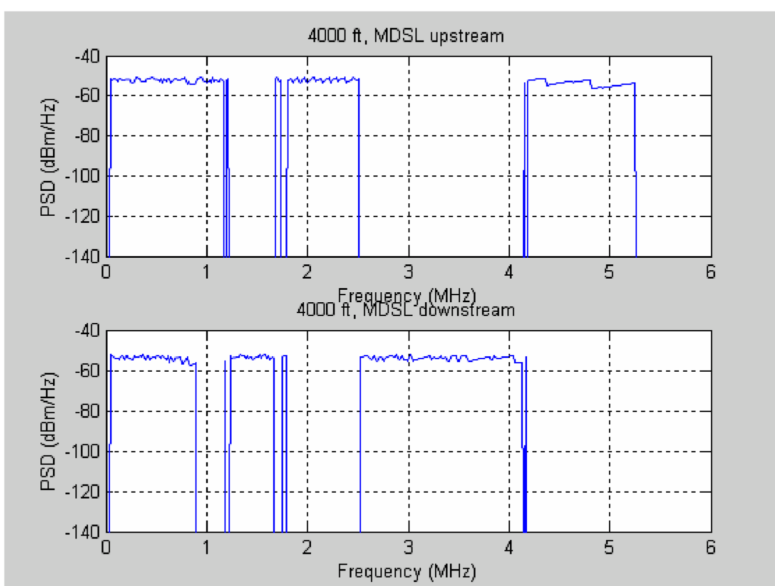

Figure 15.9: Example PSDs.

as they could everywhere. However, under a power constraint for the other $U-1$ users individually, the problem shifts to a game in which user u tries to maximize their data rate, while all the others try to minimize this same data rate of user u with offensive choices of power spectra that however must satisfy the indivdual power constraints. The other users employ a strategy to reduce this one victim user's data rate and thus create a "worst case." Mathematically, this problem is written as

$$\max_{R_{\boldsymbol{xx}}(u)} \min_{R_{\boldsymbol{xx}}(i \neq u)} \log_2 \frac{|H_{uu} R_{\boldsymbol{xx}}(u) H_{uu}^* + R_{noise}(u)|}{|R_{noise}(u)|} \quad , \tag{15.11}$$

$$ST: \tag{15.12}$$

where as usual

$$R_{noise}(u) = \sum_{i \neq u} H_{ui} R_{\boldsymbol{xx}}(i) H_{ui}^* + R_{\boldsymbol{nn}}(u) \quad . \tag{15.13}$$

The constraints are that the user autocorrelations are valid (positive semi-definite) matrices with bounded traces (or sum traces when formulated in frequency domain for set of tones) by the energy constraints. Power spectral density constraints may also be applied to each tone in "loading" for the worst-case interference.

The analysis of Brady's method is somewhat involved and beyond the scope of this version of EE479. Nonetheless, we have provided a software routine in Mat lab that solves the above problem (all users have $L_x = L_y = 1$.

The program has 4 inputs (with user 1 considered to be the victim and all others to be the offenders):

1. H the $U \times U \times N$ tensor of all the channel responses
2. P a $U \times 1$ vector of individual energy per-user constraints
3. Sigma is the $U \times 1$ vector of noise variances
4. Gap is the linear scale gap

and 4 outputs

1. R is the number of bits per symbol for the victim user 1
2. Y is the $(U-1) \times N$ matrix of PSDs of the offending modems
3. X is victim user 1's power spectral density
4. int_profile is the sum of offending users' channel output spectra into user 1.

The program provided by Mark Brady is listed here:

```
function [Rate Y X int_profile]=wci(H,P,Sigma,Gap)
%function [Rate Y X int_profile]=wci(H,P,Sigma,Gap)
%Compute the worst-case interference for IC user 1
%Inputs: H is a UxUxN matrix of channel gains
%          where H(m,p,n) is channel from user p into user m on tone n
%       P is a Ux1 vector of power constraints.
%       Sigma is a Nx1 vector of AWGN noises (per tone noise)
%       Gap is the Gap-to-Capacity in LINEAR scale (not dB)
%Outputs:   Rate is the guaranteeable rate under WCI
%          Y is the worst-interference inducing power allocations
%          X is the victim modem response to the worst-interference
%          int_profile is WCI interference profile
%Restrictions:     *H of user 1 should not be zero for all tones
%                 *H must have at least 2 users
%                 *P must have each element strictly positive
%rev mhbrady 9/23/05
```

```
%Test the input dimensionality
Rate = 0;
Y = 0;
X = 0;
int_profile = 0;

TOL = 1e-20;  % Channel gains less than this are treated as 0

try
    TONESRAW = size(H,3);

    if size(Sigma) ~= [TONESRAW 1]
        disp('Invalid input dimensions for Sigma')
        return
    end
    if or(prod(size(Gap)) ~= 1,Gap < 1)
        disp('Invalid Gap')
        return
    end
    if or(size(P,1) ~= size(H,1), size(P,2) ~= 1)
        disp('Invalid input dimensions of P')
        return
    end

    if size(H,1)~=size(H,2)
        disp('Invalid input dimensions. H must be square (per tone)')
        return
    end


catch
    disp('Invalid input dimensions, or not enough inputs.')
    return
end


%Get power gains
Hsq = H .^ 2;
%Initialize interference structure
A = zeros(size(H,1),size(H,3));
Hsq_new = zeros(size(H,1),size(H,2));
Sigma_new = [];
%Extract TONESRAW where victim has gain of < TOL
pos = 1;
ontones = [];
for n=1:TONESRAW
    if(H(1,1,n) >= TOL)
        Hsq_new(:,:,pos) = Hsq(:,:,n);
        Sigma_new(pos,1) = Sigma(n);
        ontones = [ontones n];
        pos = pos+1;
    end
end
```

```
TONES = size(Hsq_new,3);

%Do normalization
normlz = Hsq_new(1,1,:);
normlz = reshape(normlz,prod(size(normlz)),1);
for intr=2:size(H,1)
    Htemp = Hsq_new(1,intr,:);
    Htemp = reshape(Htemp, prod(size(Htemp)),1);
    A(intr,:) = Gap*(Htemp ./ normlz)' ;
end

Sigma_norm = Gap*(Sigma_new ./ normlz)';

CAP = 2*repmat(P,1,TONES);
ABSTOL = .000001;

Atilde{1} = A;
for u=2:size(H,1)
    Atilde{u} = 0*A;
end

w = zeros(size(H,1),1);
w(1) = 1;

Sigma_norm2 = [Sigma_norm'...
    repmat(Sigma_norm',1,size(H,1)-1)];

[Xback, Yback, LB_cp]=linw(Atilde,Sigma_norm2',P,CAP,w',ABSTOL,0);

Rate = LB_cp;
Yb = Yback(2:size(Yback,1),:)';
Xb = Xback(1,:)';

%Do the writeback
int_profile2 = (normlz' / Gap) .* (sum(A .* Yback,1)+Sigma_norm);
int_profile2 = int_profile2';
int_profile = zeros(size(H,3),1);
int_profile(ontones) = int_profile2;

Y = zeros(size(H,3),size(H,1)-1);
X = zeros(size(H,3),1);

Y(ontones,:) = Yb;
X(ontones,:) = Xb;
```

The wci program was used to generate worst case noises for the near/far downstream ADSL and upstream VDSL examples in Figures 15.10 and 15.11. (The acronym PBO in these curves corresponds to "power back off," which corresponds to a method where near-user crosstalk is controlled to be the same level in power as the longest line at the line at same distance as the present line's receiver.) The IW plot shown is an RA IW. The power limit on the near modem was reduced by the amount shown on the horizontal axis in both plots, effectively simulating a "fixed-margin" effect as the data rate reduces for the near connection while the rate for the far connection increases. In these figures, the wci is close to the RA WF spectrum allocation in all cases. This suggests that while the best operating point is somewhere near the middle of each plot with FM IW (and this may be a considerable improvement with

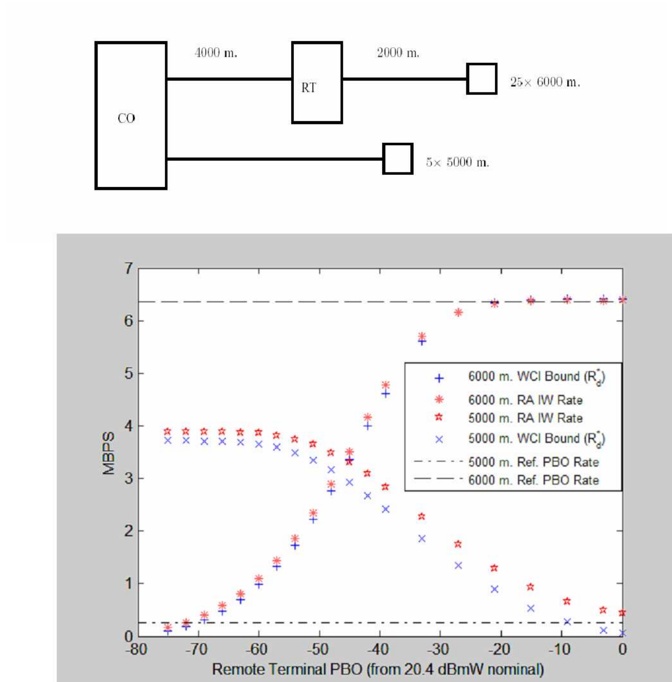

Figure 15.10: ADSL Worst Case Noise Example

respect to current systems that operate near the right-hand side of the plots), that perhaps better yet allocation of spectra would be possible.

Sections 15.3 and 15.4 address algorithms that presume some central allocation of spectra (but no successive decoding) to effect further improvements in data rate for the situation, moving away from the worst-case spectra.

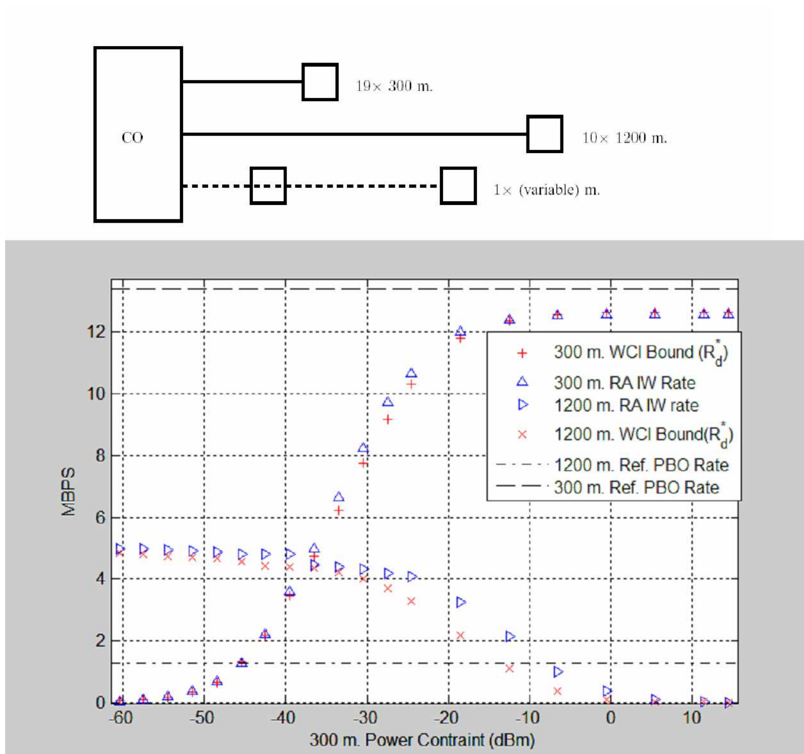

Figure 15.11: VDSL Worst Case Noise Example

15.3 Centrally Controlled Spectrum Allocation without Successive Decoding

This section specifically allows central control and focuses on the CCIC, but no successive decoding is presumed implemented by any of the users' receivers. Subsection 15.3.1 reviews Cendrillon's "Optimal" spectrum balancing (OSB), a method that is optimum for a synchronized set of DMT systems with an infinite number of tones and provides a guideline for more feasible central controlled systems without successive decoding. The complexity of OSB is exponentially high and not feasible for most problems even when used with a finite number of tones. Iterative Spectrum Balancing (ISB) replaces the most intensive step of OSB with an approximation as in Subsection 15.3.2 and often produces at a lesser (but still high) complexity a result. Papandriopoulos' SCALE method for the **IC** reduces complexity to a level comparable to the **MAC** minPMAC program of Section 13.5, Subsection 15.3.3. Additionally, the SCALE approximations provide a nice analogy with iterative water-filling, leading ultimately to the iterative multi-level water-filling methods of Section 15.4 that will obtain essentially the highest level of performance at a cost essentially no greater than that of IW and with a highly distributed implementation (although a very small amount of central coordination is used).

15.3.1 Optimum Spectrum Balancing (OSB)

The Vector DMT system is again presumed in the development of OSB so that all users are assumed to somehow use the same sample and symbol clocks in implementation. With central control, the possibility of synchronization is perhaps feasible. Even if synchronization is not quite perfect, the methods of this subsection provide theoretical guidelines and are not intended for practical implementation, making the presumption of perfect synchronization in the **IC** more palatable for the development of bounds.

As in the **MAC** , a weighted rate sum is maximized with the selection of non-negative weights allowing the trace of an achievable region.

The basic problem may be stated[6]

$$\max_{\{R_{\boldsymbol{xx}}(u,n)\}} \quad \sum_{u=1}^{U} \theta_u \cdot b_u \tag{15.14}$$

$$ST: \quad 0 \le \sum_n \text{trace}\left\{R_{\boldsymbol{xx}}(u,n)\right\} \le \mathcal{E}_{u,max} \;\; u = 1, ..., U \tag{15.15}$$

The relationship between user bit distribution and autocorrelation matrices remains as

$$b_u = \sum_n \log_2 \frac{\mid H_{uu,n} R_{\boldsymbol{xx}}(u,n) H^*_{uu,n} + R_{noise}(u,n) \mid}{\mid R_{noise}(u,n) \mid} \quad . \tag{15.16}$$

OSB methods are usually derived only for the case $L_x = L_y = 1$; while the generalization to vector channels may be straightforward, it is tedious and this version of these notes will not further pursue the vector generalization. Thus, it is alright to set $\mathcal{E}_{u,n} = R_{\boldsymbol{xx}}(u,n)$ in the present sequel. As in Section 13.4, the Lagrangian can be formed and written as a sum of tonal components defining

$$L_n(R_{\boldsymbol{xx}}(u,n), \boldsymbol{b}_n, \boldsymbol{w}, \boldsymbol{\theta}) = \sum_n w_u \cdot \text{trace}\left\{R_{\boldsymbol{xx}}(u,n)\right\} - \theta_u \cdot b_{u,n} \quad , \tag{15.17}$$

then the overall Lagrangian is, with user energy constraints being the diagonal elements of $\mathcal{E}_{vec}$,

$$L = \sum_{u=1}^{U} \left[\sum_n L_n \right] - w_u \cdot \mathcal{E}_{u,max} \quad . \tag{15.18}$$

The Lagrangian problem is not convex because each user depends on all other users' spectra. However, it does have a solution. The achievable rate region for no successive decoding can be traced by using

[6] This problem is the same as maximizing one users' rate while all ohters are lower-bounded at some desired rate each.

all convex combinations such that $\sum_{u=1}^{U} \theta_u = 1$ with $\boldsymbol{\theta} \succeq \mathbf{0}$ and the implied constraint $\boldsymbol{w} \succeq \mathbf{0}$. Since there is no use of successive decoding, there is no order for a decoder. Thus, $\boldsymbol{\theta}$ does not in the OSB case determine an order. Furthermore, the relation in (15.16) can be rewritten with a gap for scalars so that (with $\text{SNR}_{u,n} = \frac{|H_{uu,n}|^2 \cdot \mathcal{E}_{u,n}}{R_{noise}(u,n)}$

$$b_u = \sum_n \log_2 \left(1 + \frac{\text{SNR}(u,n)}{\Gamma}\right) \quad . \tag{15.19}$$

Since each user is essentially a single user coding against all others as noise, then the gap approximation directly applies (unlike with successive decoding or in the **MAC** or **BC**).

The Lagrangian also applies in the case where the user data rates are provided and the sum of energies is minimized, in which case the trailing term in (15.18) becomes $\theta_u \cdot b_u$ instead of $-w_u \cdot \mathcal{E}_{vec}(u)$ and the Lagrangian is then minimized instead of maximized. Equivalently, the doubly constrained problem can be checked for energy-constraint satisfaction at any given $\boldsymbol{b}$ (the admission problem). This check can be used to generate a new $\boldsymbol{\theta}$, which then can be subsequently used again in the original problem. Thus the algorithm has two steps:

1. minimize L_n for fixed $\boldsymbol{w}$ and $\boldsymbol{\theta}$ by using the known capacity relation in (15.19).
2. Optimize using sub-gradient descent (or the elliptical method) for the value of $\boldsymbol{w}$ (or both $\boldsymbol{\theta}$ and $\boldsymbol{w}$ in the admission-problem context).

The second step is as in the case of the **MAC** and follows the same (elliptical or sub-gradient) algorithms. The first step is not the same because of the interdependencies of the $R_{\boldsymbol{xx}}(u,n)$ without successive decoding. Thus, exhaustive search of all possible energy settings for all users is required.

That first step can be evaluated for all energy vectors up to the maximum. Defining

$$M = \frac{\max_u \mathcal{E}_{vec}(u)}{\Delta \mathcal{E}} \tag{15.20}$$

for some energy search increment $\Delta\mathcal{E}$, then M^U evaluations of L_n are necessary for each tone. This is a high complexity for more than 2 or 3 users. The actual complexity has order $O(NUM^U)$ because each $R_{noise}(u,n)$ calculation itself requires U computations.

The second step can use sub-gradients (let $\boldsymbol{E}_{max}$ be a vector of the diagonal elements of $\mathcal{E}_{vec}$ and $\boldsymbol{E}_n$ the vector of energies for the users on any tone n)

$$\Delta \boldsymbol{b} = \boldsymbol{b}_{min} - \sum_n \boldsymbol{b}_n \tag{15.21}$$

$$\Delta \boldsymbol{E} = \boldsymbol{E}_{max} - \sum_n \boldsymbol{E}_n \quad . \tag{15.22}$$

These sub-gradients can be used in an elliptical search procedure as in Section 13.5, or they can be used for direct update of the Lagrange multipliers according to

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \epsilon \cdot \Delta \boldsymbol{b} \tag{15.23}$$

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \epsilon' \cdot \Delta \boldsymbol{E} \quad . \tag{15.24}$$

Typically, the elliptical procedure is necessary for reasonable convergence in practice, where the initial condition for $\boldsymbol{\theta}$ (and $\boldsymbol{w}$ in the admission problem) is chosen in the first quadrant.

15.3.2 Iterative Spectrum Balancing (ISB)

Iterative Spectrum Balancing (ISB) was simultaneously introduced by Yu and Liu and by Cendrillon in May 2005. ISB attempts replacement of the exhaustive search in minimizing L_n on each tone by an iterative approximate algorithm. The basic idea in that step is to optimize each user separately in a sub step where M values of energy for that user are compared in terms of L_n values while the energies for all other users are held constant. The algorithm cycles through all users (each holding all the others constant). Convergence is assured because the method reduces L_n at each step, and usually in far less than M^U steps. Thus, the complexity reduces from $O(NU(M^U)$ in OSB to $O(NU^2M)$. By comparison, IW requires $O(NU)$ and thus is still considerably less complex yet than ISB, which however is much less complex than OSB.

15.3.3 SCALE

SCALE (Successive Convex Approximation for Low complExity) was introduced in 2005 by Papandriopoulis. This method is similar in complexity to ISB and also replaces the exhaustive search step by an iteration, but does so by bounding the Lagrangian with a convex approximation. This approximation itself is updated. The SCALE algorithm also suggests the merit of distributed loading and attempts to describe a set of minimal messages that can be passed from/to the CC to/from the users. In so developing the criterion, the relationship to iterative water-filling becomes clear in some terms of the messages that essentially then become superfluous in IW. This observation leads to the multi-water-level IW methods of Section 15.4.

The bound

$$\alpha \cdot \log z + \beta \leq \log(1+z) \tag{15.25}$$

where

$$\begin{aligned} \alpha &= \frac{z_0}{1+z_0} && (15.26) \\ \beta &= \log(1+z_0) - \frac{z_0}{1+z_0} \cdot \log(z_0) \quad , && (15.27) \end{aligned}$$

holds for any $z_0 > 0$ with the bound[7] becoming more tight as $z \to z_0$. Use of (15.25) in the optimization criterion leads to

$$L = \max_{\{R_{\boldsymbol{xx}}(u,n)\}} \sum_n \sum_{u=1}^{U} \theta_u \cdot \left[\alpha_{u,n} \cdot \log_2 \underbrace{\frac{| H_{uu,n} R_{\boldsymbol{xx}}(u,n) H^*_{uu,n} |}{| R_{noise}(u,n) |}}_{SIR(R_{\boldsymbol{xx}}(u,n))} + \beta_{u,n} \right] \tag{15.28}$$

$$ST: \quad 0 \leq \sum_n \text{trace}\left\{ R_{\boldsymbol{xx}}(u,n) \right\} \leq \mathcal{E}_{vec}(u) \;\; u = 1, ..., U \quad . \tag{15.29}$$

When $L_x = L_y = 1$, then the $R_{\boldsymbol{xx}}(u,n)$ terms are replaced by the scalars $\mathcal{E}_{u,n}$. The substitution

$$\tilde{\mathcal{E}}_{u,n} \stackrel{\Delta}{=} \log \mathcal{E}_{u,n} \tag{15.30}$$

leads to a convex Lagrangian that can be optimized by descent or fixed-point algorithms. In fact the exhaustive-search step transforms to an interaction by setting the derivative of the tonal Lagrangian equal to zero so that

$$\frac{\partial L_n}{\partial \tilde{\mathcal{E}}_{u,n}} = 0 = \theta_u \cdot \alpha_{u,n} - \mathcal{E}_{u,n} \cdot \left(w_u + \sum_{i \neq u} \alpha_{i,n} \cdot \theta_i \cdot \frac{|H_{iu,n}|^2}{R_{noise}(i,n)} \right) \quad , \tag{15.31}$$

leading to the loading energy equation

$$\mathcal{E}_{u,n} = \frac{\theta_u \cdot \alpha_{u,n}}{w_u + \sum_{i \neq u} \alpha_{i,n} \cdot \theta_i \cdot \frac{|H_{iu,n}|^2}{R_{noise}(i,n)}} \quad . \tag{15.32}$$

The solution depends on the power spectra of the other users through $R_{noise}(i,n)$ terms so a iteration is necessary where the other users powers are held fixed and actually the power of user u is fixed at the previous value for calculation of each of the $U-1$ $R_{noise}(i,n)$ terms. This iteration converges because

[7] The bound holds for any common base of the logarithms involved.

at each step for each user, the value of the Lagrangian on the tone of interest reduces. Thus, SCALE greatly simplifies the internal step, but perhaps more important is the concept of message passing tacit in SCALE. Namely, SCALE defines a message that comes from all users to the CC:

$$m_{u,n} = \frac{\theta_u \cdot \alpha_{u,n}}{R_{noise}(u,n)} \quad . \tag{15.33}$$

Using these centrally received messages, the CC computes some control messages for return transport to the users

$$\tilde{m}_{u,n} = \sum_{i \neq u} |H_{iu,n}|^2 m_{i,n} \quad . \tag{15.34}$$

The local loading algorithm then computes the energy directly as

$$\mathcal{E}_{u,n} = \frac{\theta_u \cdot \alpha_{u,n}}{w_u + \tilde{m}_{u,n}} \quad . \tag{15.35}$$

Thus, the exhaustive-search step of OSB on each tone is replaced by an iteration of single-steps for each user to compute each its best energy at that tone using the messages received from the CC. The values of α and β can be tightened for each step by setting $z_0 = SIR\left(R_{\boldsymbol{xx}}(u,n)\right)$ from the previous recursions. Perhaps of greater interest yet though is that by returning to IW and using the same α and β approximations that maximization over $R_{\boldsymbol{xx}}(u,n)$ for each user with all others deemed constant and not part of the optimization leads to the loading step

$$\mathcal{E}_{u,n} = \frac{\theta_u \cdot \alpha_{u,n}}{w_u} \tag{15.36}$$

and so the message is zero in IW (i.e., a DCIC). This observation leads to “multi-water-level” methods in Section 15.4.

15.4 Multi-Level (Iterative) Water Filling

The optimization methods discussed in Sectin 15.3 provide nice bounds, and increasingly less complex methods to compute those bounds, for the interference channel without iterative decoding. However, there are enormous realization difficulties with all these methods:

1. the OSB family of central spectra control all assume synchronization (as with the **MAC** or **BC**) so that crosstalk only occurs between systems independently on each tone. Such synchronization is unlikely and thus the crosstalk will be a function of other adjacent tones (and the spectra of unsynchronized tones only falls as $1/f$ so the other tones' crosstalk will be significant – even windowing can ony reduce this effect slightly as in Section 4.9).

2. In unsynchronized systems, the power-spectral density roll-off prevents very large reductions in power-spectral density on adjacent tones, which is exactly the type of spectra that OSB methods produce. Indeed most transmission standards restrict the transition bands of power-spectral density programmability so that arbitrarily large spectrum reduction is not possible.

3. Central control of spectra (that is central bit and gain swapping) is not feasible from a speed of reaction standpoint in most systems (without significant loss in data rate caused by control channels for feedback and feedforward of channel state information and transmitter-state controls).

4. the reaction of the modems with distributed algorithms to adapt to channel changes is heavily restricted in the OSB family of algorithms.

These difficulties encourage the development of more distributed autonomous spectral balancing among the multiple users.

The SCALE algorithm's proximity to iterative water-filling suggests a much more realistic and robust approach to the interference channel. An algorithm introduced first by Lin and Su[8] can be modified and interpretted as a multi-level water-filling algorithm, and is here recognized to approximate often the OSB-family like results. The basic idea is to water-fill to different levels in two or more frequency bands, the difference in water-filling levels is determined by the algorithm but some bands are preferred for loading with a higher water level simply by noting that the allowed power spectral density mask (typically provided infrequently by control system or by standard) suggests that that band should be preferred in allocation of bits in a LC-style loading method. Section 15.4.1 describes this multi-level water-filling in more detail. Section 15.4.2 illustrates the many advantages of this method with some examples. In all cases, the method closely approximates OSB achievable regions but with very low complexity loading, and even more importantly implementation of the loading within the modems themselves with no need for central control.

15.4.1 The ML IW method

Equation 15.36 illustrates that OSB-style optimization essentially reduces the energy allocated (and thus data rate) to bands of high observed crosstalk. Lin and Su introduced the concepted of a fully distributed modification of Chapter 4's Levin-Campello (LC) essentially swaps bits from the best tones to the worst loaded tones until PSDMASK's are saturated on the worst tones. On a channel with no crosstalk, or from a purely single-user perspective, this is clearly suboptimal. However, it recognizes that the most heavily loaded tones are also the mostly like to be in frequency regions that crosstalk[9]. This resonates with earlier observations that some bands are preferred for loading on short lines[10] The Lin/Su method unfortunately can be sensitive to chanes in noise per say because it will always cause a line to operate with lowest acceptable margin while the Cioffi/Mohseni method needed some mechanisum for saying the "amount of preferential loading."

[8] "Distributed DSM in DSL," *IEICE Transactions on Communications*, May 2007.

[9] "Most likely" is not an absolute, but Lin/Su were assuming that "short" lines should use higher frequencies to reduce their crosstalk into longer lines that must use low frequencies.

[10] See, Cioffi and Mohseni, December 2003 ATIS T1E1.4 Contribution "Preference in Water-filling with DSM", Contribution 321R1.

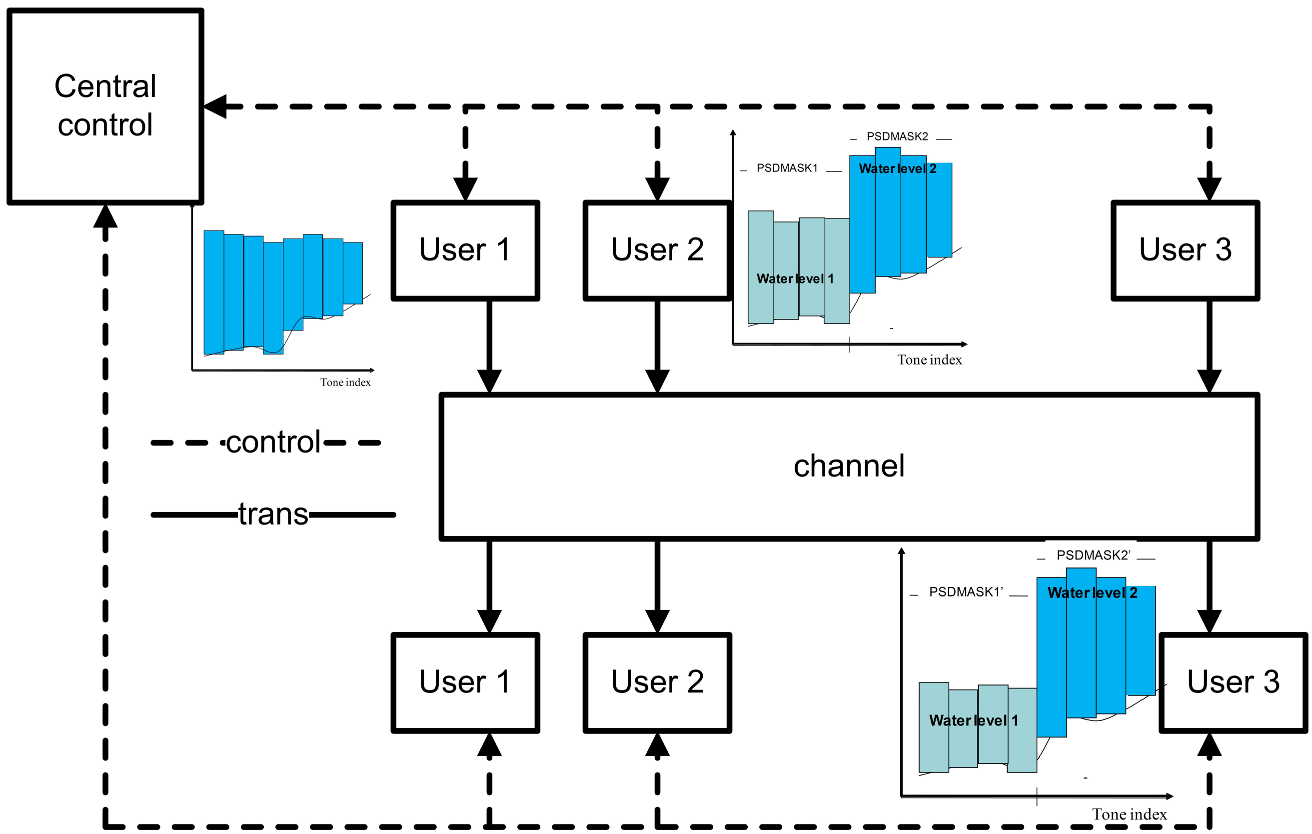

Figure 15.12: Illustration of 3-user two-band multi-level water filling.

Ultimately, the amount of need not truly be sent, but instead a separation frequency between a band of tones in which loading to the maximum level is desired more for politeness and other bands that are preferred less for politeness. Figure 15.12 illustrates the basic concept in that a control unit, or via some agreement by standard, users that can easily meet their rate targets are registered and informed to be extra polite. These users will water-fill in two bands with different water levels. Users that have no ability to be polite are not so asked to be extra polite.

An extra-polite user might have two water-filling bands (the extension to more than two bands is trivial):

$$\lambda_{u,1} = \mathcal{E}_{u,n} + \frac{\Gamma}{g_{u,n}} \forall n \in \mathcal{N}^1 \tag{15.37}$$

$$\lambda_{u,2} = \mathcal{E}_{u,n} + \frac{\Gamma}{g_{u,n}} \forall n \in \mathcal{N}^2 \quad . \tag{15.38}$$

The two bands $\mathcal{N}_1$ and $\mathcal{N}_2$ are determined by a control center that would tell a user to be exta polite, and convey the two bands (more typically the cut-off frequency between the two bands). The modem itself determines the two water-filling levels $\lambda_{u,1}$ and $\lambda_{2,u}$ via LC loading in each (of course including any appropriate power-spectral density limits with the usual infinite added-bit cost for any resultant power-spectrum that would exceed such masks). The band with the higher index 2 is preferred for loading and its water-level is determined by first solving an overall water-filling problem (with one level), and then moving bits one-by-one from band 1 to band 2 until any PSDMASK constraints in band 2 would be violated (or until the overall power constraint would be exceeded). Band 2 will be then water-fill (approximately to the degree that LC approximates water-filling) with a higher water-filling level. Band 1 will also still be water-filling but with a lower water-filling level. The cut-off frequency is important. If a very small amount of central control is allowed, then the controller would run the multi-level IW for each user for various estimates of the cut-off frequency (or exhaustive search) and then set each user's cut-off accordingly. The modems otherwise retain full automony. If noises change such that two different water levels cannot be maintained, then the simply algorithm degnerates into normal water-filling (no bits can be moved from band 1 to band 2).

The concept easily extends to 3 or more bands with the highest indexed band being filled first, then the next highest index and so forth. A simply method to communicate preferences is simply to have slight differences in the power-spectral density (PSD) mask that is communicated to all users (via central control or even by standards). A change in PSD mask from one tone n to $n+1$ simply implies that the preference index changes and the band with higher PSD mask should have a higher index. Such a method leaves all bands water-filling (and since we know IW converges under wide conditions, then this ML IW will also converge in those same situatinons). Inspection of the final results for power spectra in Sectin 15.3 certainly confirms that a simple cut-off could easily have been used to obtain the same results with IW. Thus, the only central control required is an indication to the user to load extra polite favoring those bands for which its PSDMASK is higher. The PSDMASK can be preset in all modems or might possibly have been distributed in a quasi-static upon-start type initialization within the network.

15.4.2 ML IW examples and results

The key advantages of the ML IW approach are:

1. distributed low-complexity implementation (no OSB, dual-decomposition, SCALE or other high-complexity algorithm is used. Each user implements water-filling. A central controller can implement water-filling for a few choices of cut-off and select the one that leaves best achievable rate regions. There are no convergence issues, choice of thresholds, choice of elliptic versus sub-gradient, etc complexities even at the controller if one exists.

2. The individual user modems retain the ability to react (via swapping of bits and/or gains) to changes in the noise or channel. Thus, rapid direct reaction to changes allows robust operation in the presence of any kind of situation not originally anticipated by the controller and/or modems.

3. OSB's tacit "all-are-synchronized" assumption is no longer necessary. The modems react according to the actual noise present and not some presumed synchronized-crosstalk presumption in an optimization algorithm.

4. As we shall see, the performance of ML IW matches OSB.

A few examples will illustrate the advantages.

A two-user DSL simulation is appears in Figure 15.13. A noise floor of -140dBm/Hz models background noise. Four RT (remote terminal) DSL signals constitute a strong interference to user 2.

Figure 15.14 provides the optimal spectrum levels for this channel. From this example, it is clear that the masks are very sharp in transition and nearly impossible for implementation. They also illustrate the basic concentration of short-line energy at higher frequencies. The crosstalk transfer function for the channels into one another used here was

$$| H_{ij}(f) |= 9 \times 10^{-20} \cdot (1/49)^{0}.6 \cdot l \cdot f^{2} \cdot | H_{ii}(f) |^{2} \quad , \tag{15.39}$$

(where l is the length of the line in 1000's of feet) which increases with frequency, so all lines would experience dramatic crosstalk at higher frequencies in addition to the typical attenuation with frequency of the lines. Nonetheless, the shorter lines best occupy the very lowest frequencies where the crosstalk is low and the very highest frequencies where they do not inflict harm on the high frequencies that could not be used by the long line.

By contrast, the use of ML IW would place 3 bands, low, medium, and high where low and high can have the same PSDMASK levels and the medium band should have slightly lower PSDMASK. The low and high bands are then preferred on the 4 short loops. The cut-offs are roughly 300 kHz and 700 kHz.

Figure 15.15 shows the ML IW and OSB achievable rate regions for the sitauation in Figure **??**. These two regions are the two largest, which are virtually equal. IW with no PSDMASKs is shown and is the smallest achievable region. Thus, there is a large gain possible. Each of the intermediate curves

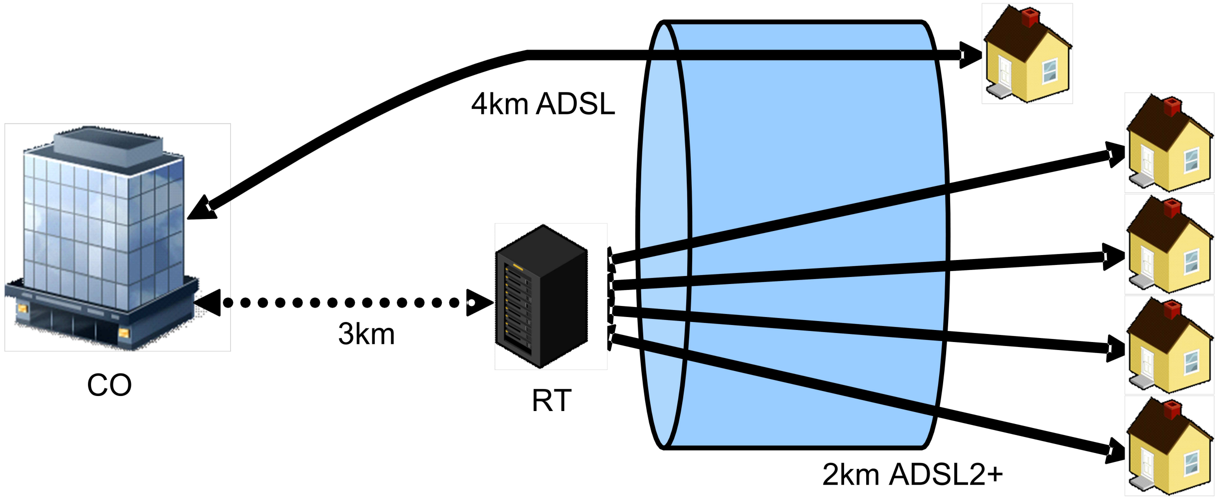

Figure 15.13: ADSL mixed binder IC example.

corresponding to relaxing the OSB PSDMASK in Figure 15.14 by successive increments of 6 dB to allow for both robustness to channel change and possibility of implementation with less sharp transition bands. ML IW has a PSDMASK in the middle band that is just 1 to 2 dB lower than the low and high bands, and thus is very feasible. These show that reasonable relaxation of the central control is highly sensitive. However, the ML IW works without need for such sensitivity.

Figure **??** shows the rate region for a 7 dB change in the noise level in the lower and middle frequency bands for ML IW, which retains essentially the same rate region. However, such a 7 dB change corresponds roughly to the curve with 6-12 dB tolerances, which lose about 20

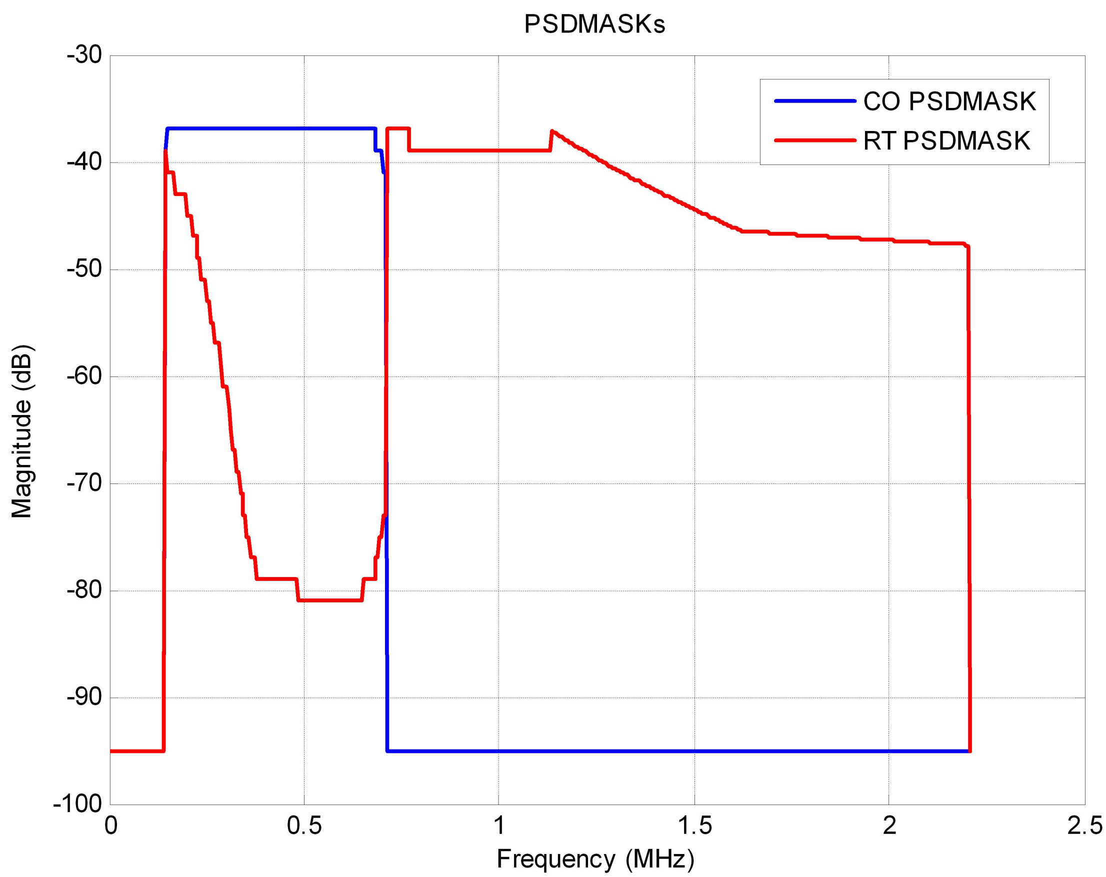

Figure 15.14: Spectra for situation in Figure 15.13.

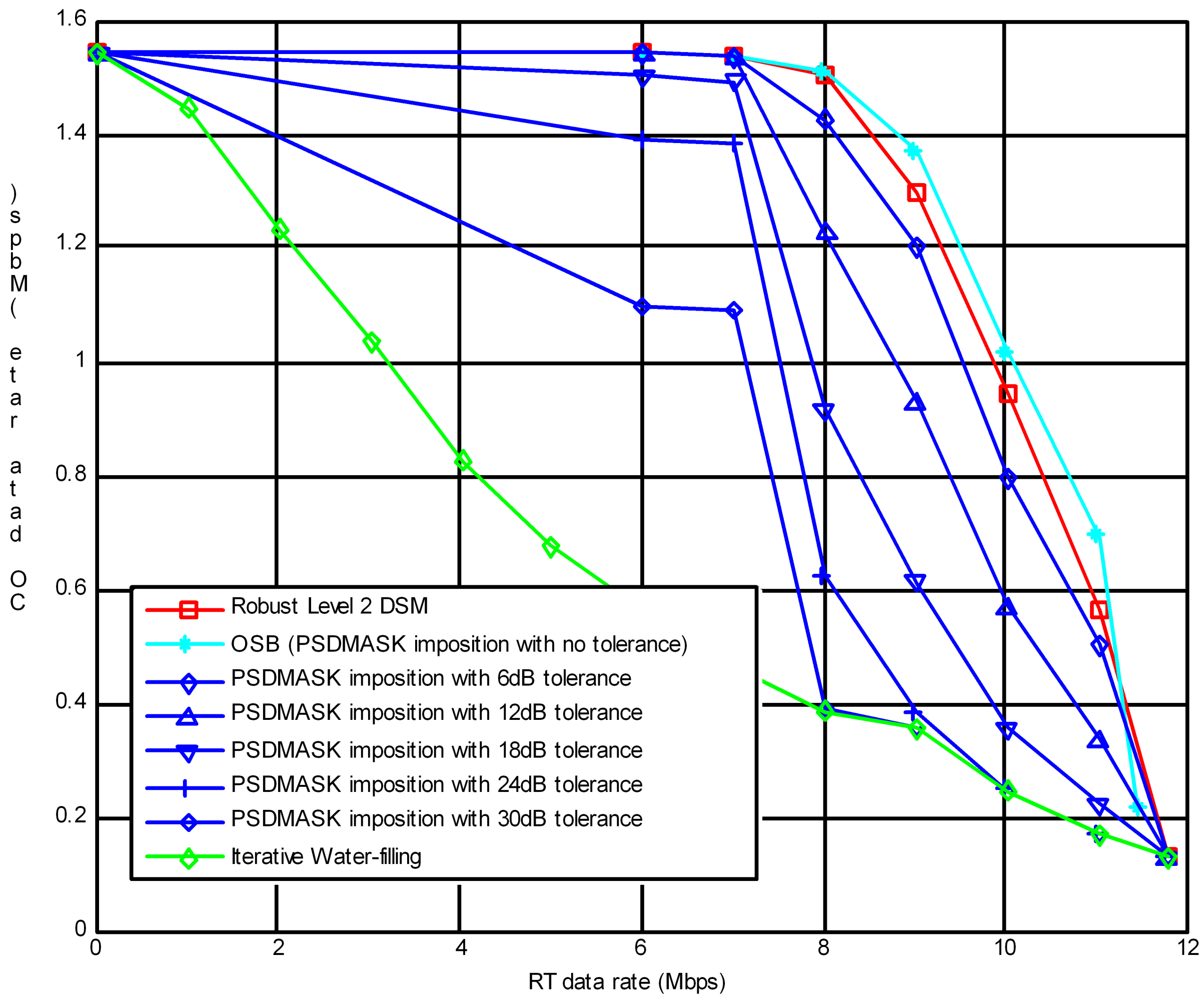

Figure 15.15: Rate Regions with various levels of tolerance on the OSB PSDMASKs.

Exercises - Chapter 15

15.1 *An interference channel*

Envision 2 WiFi 802.11(n) systems operating on adjacent floors of an office complex. Each access point has 4 transmit antennas and each of the 4 users on each floor has 2 receive antennas. The multitone system used for transmission uses (a maximum of) 64 tones with a guard period of 16.

a. Define the size of the H matrix for this IC channel and each of its constituent H_{ij} matrices if no precoding or successive decoding is used anywhere. How many users are there? Assume uplink and downlink systems all use the same frequency band. (2 pts)

b. Instead consider each of the WiFi systems independently as a downlink vector broadcast and uplink vector multiple access channels. Now repeat part a considering each system to be one user in each direction with sum rate for that system replacing the individual rates. (2 pts)

c. Which system (part a or part b) would you expect to have a larger capacity region? (1 pt)

15.2 *Achievable rate region of an interference channel with non-zero gap)*

Consider the interference channel of Section 12.4's example. Let the gap of the codes for the two users be $\Gamma > 1$ (i.e. > 0 dB).

a. Define the two MAC channels that are defined for the outputs of the two users.

b. The achievable rate region for this interference channel can be determined by the procedure of Section 15.1.2. When the gap is 0 dB, this procedure can be simplified by considering only the rate pairs achieved by successive decoding at each user for both orders and for different energy scalings for the 2 users. However, as shown in problem 1 for the scalar MAC, successive decoding is not optimum when the gap is non-zero. Develop an intersection procedure that considers the achievable rate regions of the 2 MACs simulataneously and plot the achievable rate region of the IC when the gap $\Gamma = 4$ dB.