Contents

Chapter 14

The Gaussian Vector Broadcast Channel

The Gaussian **Vector Broadcast Channel (BC)** first appeared in Chapter 12, where a single transmit signal containing U users' possibly simultaneous messages passed over several parallel (and not necessarily independent) channels to U physically distinct receivers. Chapter 12's introduction considered mainly scalar transmit signals. This chapter investigates the design of best transmitters and receivers for the **BC**. While the special-case $1 \times U$ BC of Chapter 12 admitted both a successive-decoding receiver or a precoder as a best implementation, the precoder method is more general and will apply to all **BC** 's. This precoder approach allows the construction of the **BC** 's dual-channel as a **MAC**. The dual-channel's dual-GDFE then can be used to determine the corresponding best **BC** transmit and receiver processing. The dual **MAC** channel will have each user with the same data rate and a corresponding set of input autocorrelation matrices for these user rates. The sum of the U input energies for the dual **MAC** channel will equal the energy for the **BC**. Chapter 13's design methods for maximum rate sum or individual rate points (minPMAC) can then be applied to the dual **MAC** channel's realization, and then consequently the best **BC** implmentation then also follows.

Section 14.1 introduces a vector model for the **BC** that is essentially the transpose of the model for the **MAC** in Section 13.1. Section 14.1 then revisits the precoder method from Section 12.3 and further refines that method before stating the more general **BC** form of the capacity region for the **BC**.

Section 14.2 then progresses to a discussion of worst-case noise and how it may be of use generally in the **BC** because of its GDFE feedforward-section diagonalization property, which is particularly useful in directly computing the **BC** maximum rate sum. A caution in the use of worst-case noise also occurs in the form of a simple 1×2 BC example, which is then better addressed through the concept of the scalar duality of Section 12.3. This then motivates Section 14.3 that will introduce a general form of duality using the concepts of dual channels and input deflection. The dual-GDFE of a the **BC** 's dual channel (said dual channel being a **MAC**) then allows design of the best **BC** transmitter (and somewhat trivial receiver) from the components of the dual-GDFE transmitter and receiver designs. The feedback section of the dual-GDFE will become the precoder of the GDFE for the **BC** , while the feedforward processing of the dual-GDFE becomes the transmit matrix filter for the **BC** , and the dual channel is the transpose of the original **BC** H matrix. Section 14.4 then concludes with the Vector-DMT approach for the **BC** .

The determination of the best input autocorrelation in **??** for any point $\boldsymbol{b} \in c(\boldsymbol{b})$ follows by re-using the minimum-energy-sum Mohseni methods (and software) of Section 13.4 on the appropriately defined dual **MAC** channel for any **BC**, thus generating a set of input covariances and an order (which is again the same on all tones if a VDMT implementation) for implementation of the precoder and receivers for any point in the **BC** capacity region, as well as the algorithmic generation of the capacity region.

14.1 The Vector Broadcast Channel

The Gaussian vector broadcast channel parallels Chapter 13's **MAC** in many ways, but most simply and importantly, the **BC** can be viewed as the (conjugate) transpose of a **MAC** . Subsection 14.1.1 provides this transpose model for the **BC** . Subsection 14.1.2 revisits Forney's Crypto Precoder to describe an encoder that could be used for the **BC** before restating the **BC** form of the capacity region from Chapter 12.

14.1.1 Modeling the BC

Figure 14.1 illustrates the **BC**. The single vector input $\boldsymbol{x}$ has dimensionality $L_x(N+\nu)\times 1$ and is the sum of U independent components for the Gaussian channel

$$\boldsymbol{x} = \sum_{u=1}^{U} \boldsymbol{x}_u \quad . \tag{14.1}$$

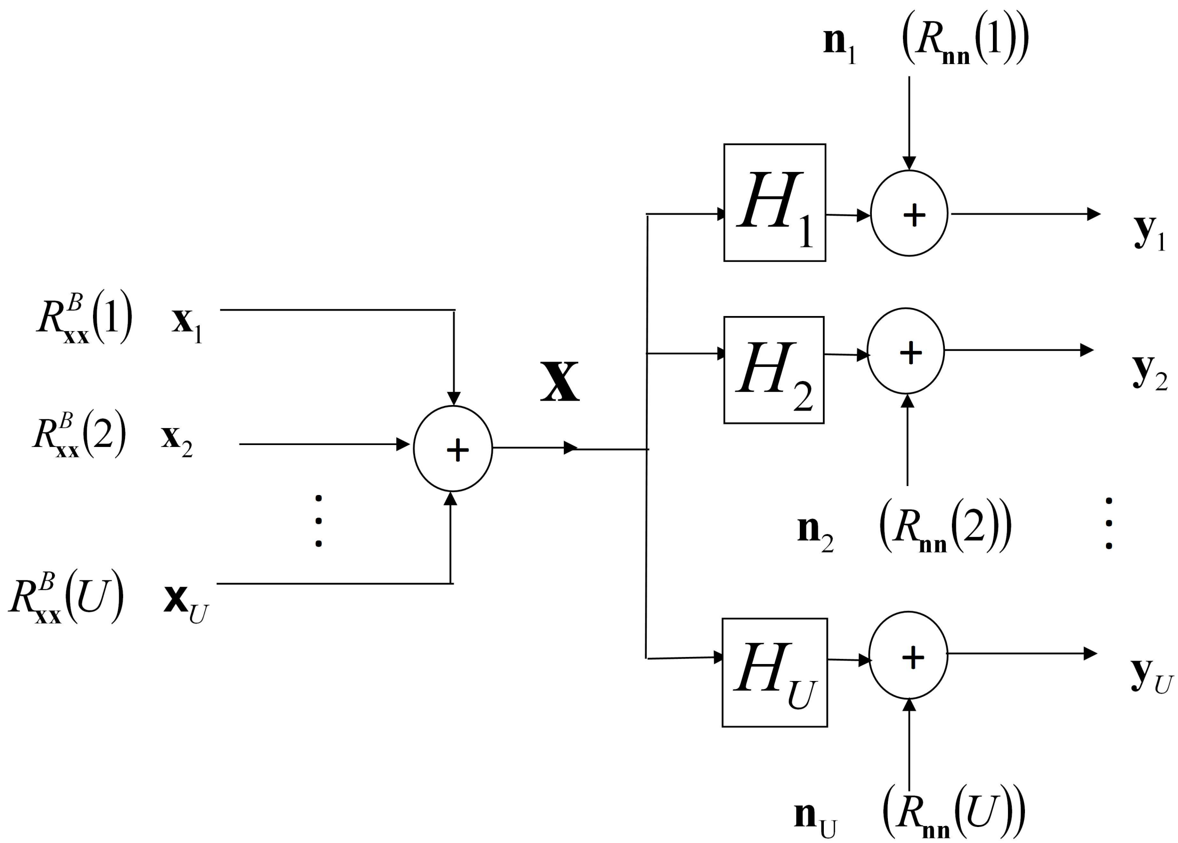


Figure 14.1: The **BC** model.

The sufficiency of linear modulation and all Gaussian users was established more generally in Chapter 12 and is presumed throughout this chapter. Each of U channel matrices H_u $u = 1, ..., U$ multiplies this same input to produce an $L_y NU \times 1$ channel output vector that satisfies

$$\boldsymbol{y} \;=\; \underbrace{\begin{bmatrix} \boldsymbol{y}_U \\ \vdots \\ \boldsymbol{y}_1 \end{bmatrix}}_{L_yNU\times 1} = \underbrace{\begin{bmatrix} H_U \\ \vdots \\ H_1 \end{bmatrix}}_{\substack{H \\ L_yNU\times L_x(N+\nu)}} \underbrace{\boldsymbol{x}}_{L_x(N+\nu)\times 1} + \underbrace{\begin{bmatrix} \boldsymbol{n}_U \\ \vdots \\ \boldsymbol{n}_1 \end{bmatrix}}_{L_yNU\times 1} \tag{14.2}$$

$$\boldsymbol{y} \;=\; H\boldsymbol{x}+\boldsymbol{n} \quad . \tag{14.3}$$

The individual components $\boldsymbol{y}_n$ of $\boldsymbol{y}$ are processed individually by U physically separated receivers in the **BC** case. Each of the individual channel matrices H_u is $L_y N \times L_x(N+\nu)$, and the noise vector is also denoted more compactly as $\boldsymbol{n}$. The input autocorrelation matrix is

$$R_{\boldsymbol{xx}} = \sum_{u=1}^{U} R_{\boldsymbol{xx}}(u) \quad . \tag{14.4}$$

A single overall transmit-power constraint is expressed as

$$\text{trace}\{R_{\boldsymbol{xx}}\} \leq \mathcal{E}_{\boldsymbol{x}} \quad . \tag{14.5}$$

14.1.2 Forney's Crypto Precoder

Figure 14.2 illustrates again Forney's Crypto Precoder of Section 12.3.1. A basic interpretation of this precoder occurs for uniform input $\boldsymbol{v}_u$ over Lattice Λ_u's Voronoi region $V(\Lambda_u)$: The output $\boldsymbol{x}_u$ of the modulo device is then independent of the input $\boldsymbol{x}_u$ and of the added signal $-\sum_{i=u+1}^{U} g_{u,i}\cdot\boldsymbol{x}_i$, and more importantly this output has the same energy as the input $\boldsymbol{v}_u$, $\mathcal{E}_{\boldsymbol{v}_u} = \mathcal{E}_{\boldsymbol{x}_u}$. The result holds for $g_{u,i}$ equal to any set of coefficients. The second adder at the input to the channel is the other users' signals and

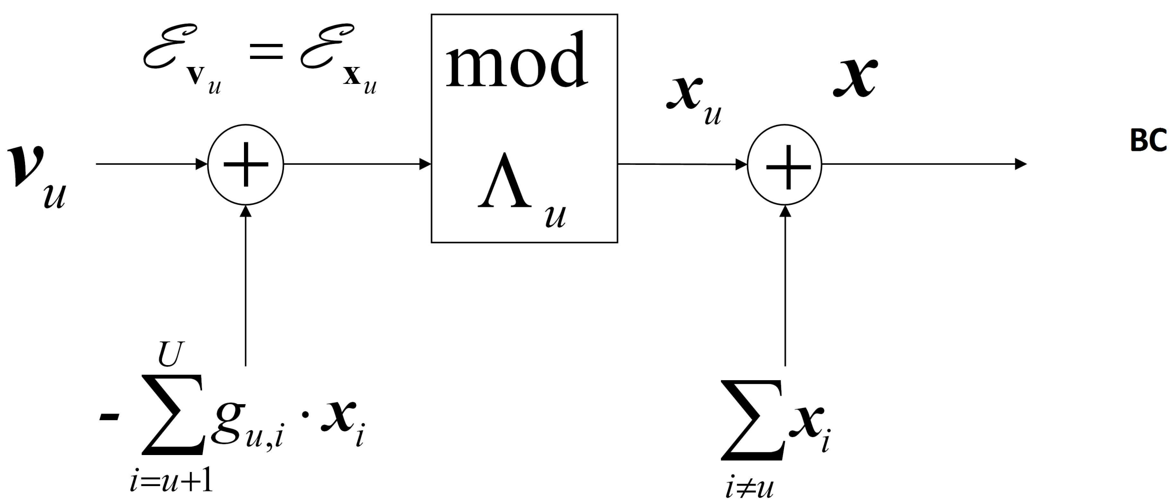


Figure 14.2: Forney's Crypto Precoder.

may be viewed as "side" information for any of the other signals that occur earlier in an ordering of users. User u views users $1, ..., u-1$ as Gaussian noise. Thus, user 1 may consequently chose its data rate as if no other crosstalk is present from other users, while user U must consider all other signals as Gaussian noise in computing its data rate.

Figure 14.3 is more explicit in showing a precoder for which each user creates a new dimension (or dimensions when $L_x N > 1$), while a linear transmit shaping matrix A combines the U inputs into an input of dimensionality $L_x(N+\nu)$., where

$$A = [A_U \; ... \; A_2 \; A_1] \quad . \tag{14.6}$$

Figure 14.3 uses the GDFE-like description with a "white" input $\boldsymbol{v}$ that has components (each on its own dimensions $\boldsymbol{v}_u$, $u = 1, ..., U$. Feedback in this precoder implementation is on each of the successive user inputs to the transmit matrix A. The modulo device in this implementation could change with each user's code in an actual system, but in the case of Gaussian codes on all users with $\Gamma = 0$ dB, this device can be viewed as forcing over an infinite number of dimensions the transmitted symbol to

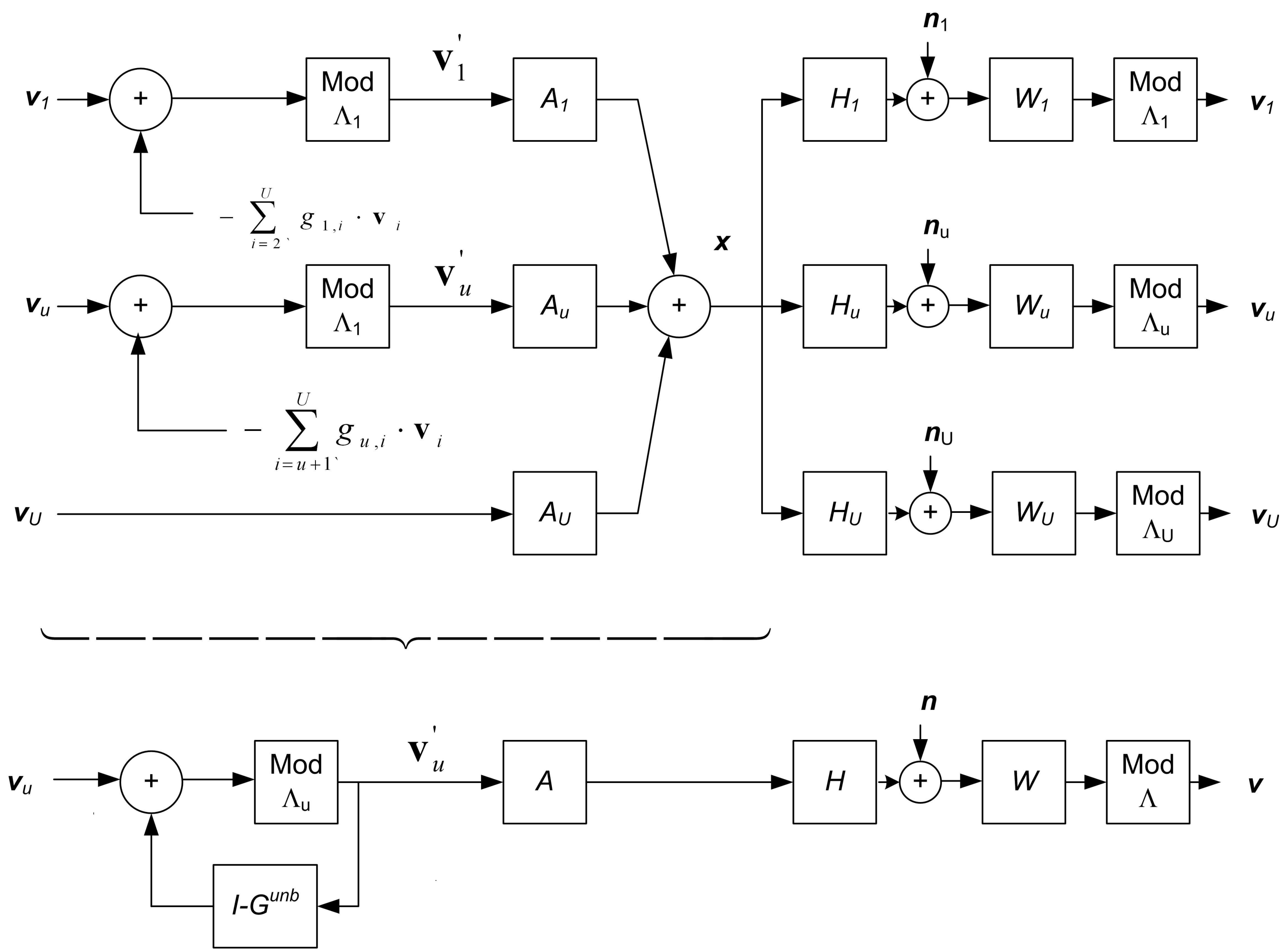


Figure 14.3: GDFE Precoder.

lie inside a Gaussian sphere with unity energy per dimension (all subsequent scaling occurring in the A matrix so that the transmit user energy components meet whatever energy assignment is desired). Equivalently, each Λ_u modulo (and associated matrix multiply A_u) operation forces a Gaussian output characterized by autocorrelation $R_{\boldsymbol{xx}}(u)$. In the scalar case of $L_x(N+\nu) = 1$, then this could be viewed as a reflection about an infinite-dimensional hypersphere of radius squared equal to the transmit energy. For a vector system, each set of $L_x(N+\nu)$ user dimensions will have its own modulo Gaussian offset that is followed by a linear filter of with characteristic $R_{\boldsymbol{xx}}^{1/2}(u)$ where any square root is allowed - such transmit matrices are generally denoted by A_u as in Figure 14.3, where $A = [A_1 ... A_U]$. Figure 14.3 also shows the U receivers that would each allow processing of the received signals independently. This may or may not correspond to a GDFE, as is investigated in Section 14.2.

In practice, the modulo device ($|Lambda_u$) in Figure 14.3 would be realized over a finite number of users and dimensions and may be some finite lattice (or even a simple slicer) for which some energy loss (beyond the non-zero gap) that occurs in a precoder like a "Tomlinson" precoder. This Chapter, like Chapter 13, does not deal with non-zero gaps thus consequently ignores finite-precoder loss in practice. Good design with good codes should make this loss negligible. With the use of a second modulo-Λ_u

device at receiver $\boldsymbol{y}_u$, those other earlier users in the order have no effect on decision for user u.

The precoder interpretation allows independent specification of each of the users maximum rates by basic mutual information expression that includes only earlier users as noise. The achievable rate region for any given set of input autocorrelation matrices is then traced by U-dimensional "boxes" of the form:

$$A'(\boldsymbol{b},\pi) = \left\{ \boldsymbol{b} \mid 0 \le b_u \le \frac{1}{2}\log_2 \frac{|\sum_{i=1}^{u} H_u R_{\boldsymbol{xx}}(i) H_u^* + R_{\boldsymbol{nn}}(i)|}{|\sum_{i=1}^{u-1} H_u R_{\boldsymbol{xx}}(i) H_u^* + R_{\boldsymbol{nn}}(i)|} \right\} \quad . \tag{14.7}$$

where the non-causal precoder is used to pre-subtract those users who are later in the order. The remaining earlier users are then considered as Gaussian noise in the denominator of the mutual-information log term. Such noise again in this chapter will be denoted $\tilde{R}_{noise}(u)$ and differs from the **MAC** in that the index of summation is constant on H_u; specifically equal to u, instead of varying with the index i. Additionally, there is a reversal of order.

Following the general capacity region of Chapter 12, the union or more exactly the convex hull over all $U!$ orders of these achievable regions is

$$A(\boldsymbol{b}) = \bigcup_{\pi}^{conv} A'(\boldsymbol{b},\pi) \quad . \tag{14.8}$$

Such an achievable region exists for all allowed input autocorrelations, and thus the capacity region is then[1]

$$c(\boldsymbol{b}) = \bigcup^{conv}_{\substack{\{R_{\boldsymbol{xx}}(u)\} \\ \sum_{u=1}^{U} \text{trace}\{R_{\boldsymbol{xx}}(u)\} \le \mathcal{E}_{\boldsymbol{x}}}} A(\boldsymbol{b}) \quad . \tag{14.9}$$

[1]A superscript of "conv" means all convex combinations or "convex hull."

14.2 Worst-Case Noise and BC Rate Sums (WCN)

Chapter 5, Section 5.5, first introduced the concept of a **worst-case noise** (WCN) for Gaussian channels. Worst-case noise occurs when the autocorrelation of the noise $R_{\boldsymbol{nn}}$ is optimized over the off-diagonal terms while the diagonal noise-power terms are held constant to minimize the mutual information. For the **BC** case, the diagonal terms are actually $L_y N \times L_y N$ block element matrices $R_{\boldsymbol{nn}}(u)$ equal to the noise autocorrelation matrices for each of the noises at the U outputs of the **BC**. The off-diagonal terms become the remaining blocks in the overall $L_y NU \times L_y NU$ noise autocorrelation matrix $R_{\boldsymbol{nn}}$. Those off-diagonal blocks are considered variable in WCN determination.

The precise generalization of WCN is then that $R_{\boldsymbol{nn}}$ that satisfies:

$$\min_{R_{\boldsymbol{nn}}} I(\boldsymbol{x};\boldsymbol{y}) = \log_2 \frac{|HR_{\boldsymbol{xx}}H^* + R_{\boldsymbol{nn}}|}{|R_{\boldsymbol{nn}}|} \tag{14.10}$$

$$ST: \quad R_{\boldsymbol{nn}}(u) \text{ fixed (BC values)} \quad . \tag{14.11}$$

The constraint amounts to holding constant the noise covariances for each "user," or for each set of dimensions for which noise must have a specified covariance. With some effort, it can be shown that the solution to this optimization problem is any matrix that satisfies

$$R_{wcn}^{-1} - \left[HR_{\boldsymbol{xx}}H^* + R_{wcn}\right]^{-1} = D \text{ block diagonal} \,. \tag{14.12}$$

The wcnoise software of Chapter 5 allows only unit-variance (or more generally identity matrix) constraints on each of the users' noise autocorrelation blocks, so then more generally only an identity matrix for $R_{\boldsymbol{nn}}$. Thus, to use that software, the individual user channels have to be pre-whitened to be

$$\bar{H}_u = R_{\boldsymbol{nn}}^{-1/2}(u) \cdot H_u \quad . \tag{14.13}$$

Each BC receiver then may have its own noise-whitening as a first processing step for its received signal to form an equivalent BC channel. This individual receiver noise whitening requires no coordination among other users' receivers.

14.2.1 WCN GDFE diagonalization

The worst-case noise GDFE analysis of chapter 5 can be summarized with some generalization in the following steps:

Given $R_{\boldsymbol{xx}}$ and H, where $\rho(H) = \rho(R_{wcn}) = UNL_y$,

1. compute R_{wcn} via [Rwcn,bwcn] = wcnoise(Rxx, H, Ly)
2. $D = R_{wcn}^{-1} - \left[HR_{\boldsymbol{xx}}H^* + R_{wcn}\right]^{-1}$, block diagonal $D \geq 0$.
3. $[0R]\,Q^* = R_{wcn}^{-1}H$ via QR factorization[2]
4. $Q = [q_1 Q_1]$, where if H is square, $Q_1 = Q$ and $q_1 = \emptyset$.
5. $UU^* = Q_1^* R_{\boldsymbol{xx}} Q_1$, Cholesky factorization where U is upper triangular.
6. $D_A = \text{diag}\{RU\}$
7. $G = D_A^{-1}RU$ (feedback section)
8. $A = Q_1 R^{-1} D_A G$ (transmit filter/matrix[3]).
9. $S_0^{-1} = D_A^{-1} D D_A^{-1} = \boldsymbol{SNR}$ (or simply $S_0^{-1}(i) = \frac{D(i)}{D_A^2(i)}$.

[2]The number of zero columns in the left matrix is zero in cases where H square. R is upper triangular. Q is an orthogonal matrix. See Example 14.2.1 for manipulation of matlab to produce precisely this filter.

[3]So $HR_{\boldsymbol{xx}}H^* = HAA^*H^*$ is a check on this value of A, and indeed for this zero-null-space construction of $R_{\boldsymbol{xx}}$, then $R_{\boldsymbol{xx}} = AA^*$, although when H is singular, only $HR_{\boldsymbol{xx}}H^* = HAA^*H^*$ may hold.

10. $W^{unb} = (\boldsymbol{SNR} - I)^{-1} G^{-*} A^* H^* R_{wcn}^{-1} = (\boldsymbol{SNR} - I)^{-1} D_A$ (diagonal feed-forward filter[4]).

11. $G^{unb} = I + \boldsymbol{SNR}(\boldsymbol{SNR} - I)^{-1}(G - I)$ (unbiased feedback section for precoder)

Figure 14.4 shows the realization of the GDFE via precoder so that the receiver consists of U sub-receivers that are not coordinated, as in a BC channel.

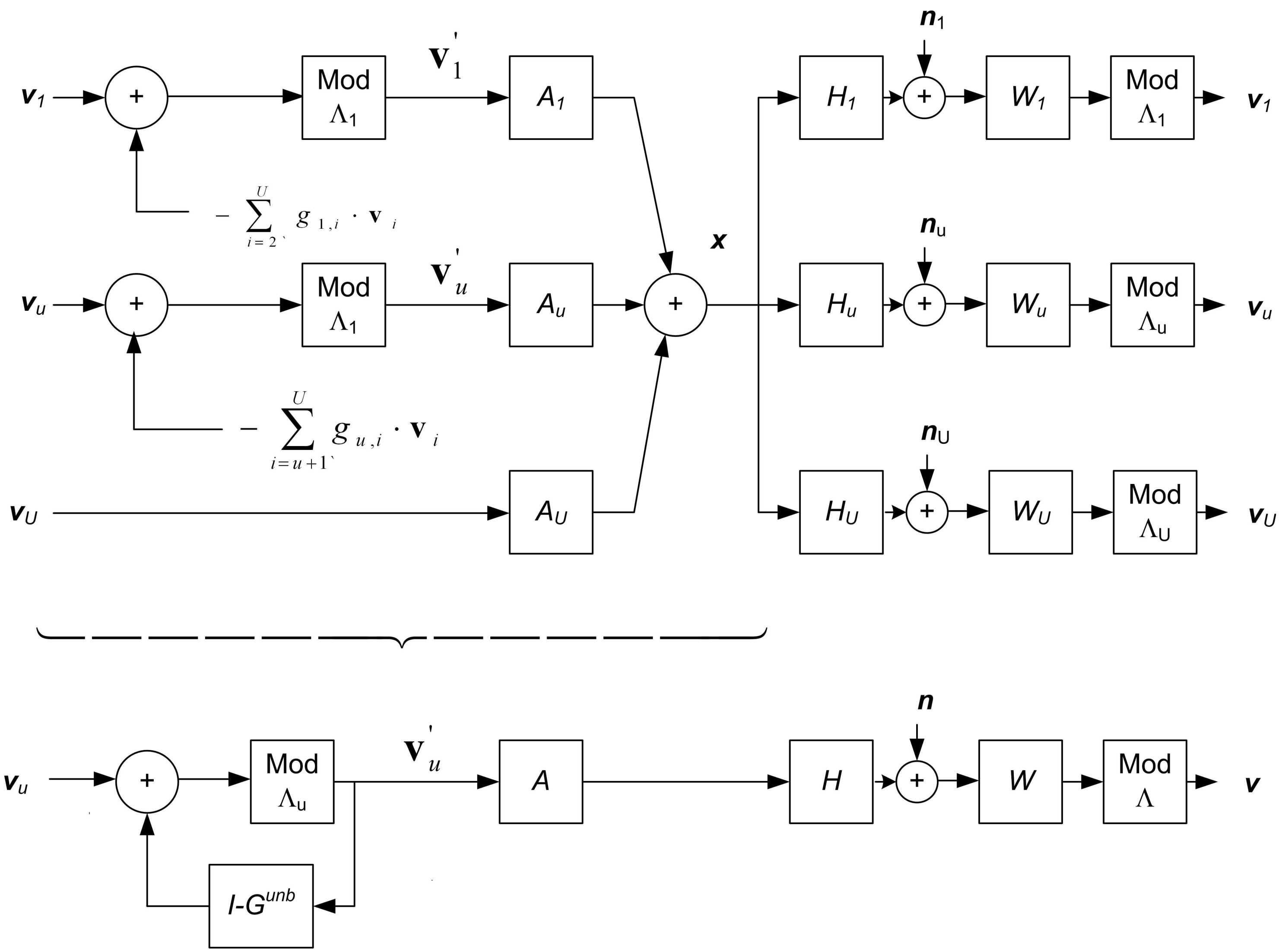


Figure 14.4: GDFE for worst-case noise with diagonal feedforward matrix - same as Figure 14.3, except that now with worst-case noise, the diagonalized receiver is known to be optimum.

EXAMPLE 14.2.1 (Worst-Case Noise for square Channel with GDFE)
```
>> H = 1.0000    0.5000
          0    1.0000    0.5000
     0.3000    0.6000    1.0000

>> Rxx = 1.0000    0.8000    0.6400
         0.8000    1.0000    0.8000
```

[4]The superscript of "unb" was used for unbiased, rather than a subscript of U as in Chapters 3 and 5 to avoid confusion with the number of users or an index value of U.

```
        0.6400    0.8000    1.0000

>> [Rwcn,bmax]=wcnoise(Rxx,H,1)

Rwcn =  1.0000    0.3871    0.4898
        0.3871    1.0000    0.6541
        0.4898    0.6541    1.0000

bmax =  1.4336

>> D=inv(Rwcn)-inv(H*Rxx*H'+Rwcn)

    0.4847    0.0000    0.0000
    0.0000    0.1985    0.0000
    0.0000    0.0000    0.4675

>> Htilde=inv(Rwcn)*H

    1.1643    0.1798   -0.2285
   -0.4787    1.0409   -0.2427
    0.0428   -0.1690    1.2707

>> J3=hankel([0 0 1]);

>> [Q,R]=qr(J3*Htilde'*J3);

>> Q=(J3*Q*J3) =

    0.9059    0.4221   -0.0334
    0.4226   -0.8967    0.1317
    0.0257   -0.1334   -0.9907

>> R=(J3*R*J3)' =

    1.1250    0.3607    0.2112
         0   -1.1030    0.3936
         0         0   -1.2826

>>  Rxxrot=Q'*Rxx*Q =

    1.6598   -0.6412   -0.8230
   -0.6412    0.5138    0.5003
   -0.8230    0.5003    0.8264

>> U=(J3*chol(J3*Rxxrot*J3)*J3)' =

    0.8621   -0.3113   -0.9053
         0    0.4593    0.5503
         0         0    0.9091

>> DA=diag(diag(R*U)) =

    0.9699         0         0
         0   -0.5067         0
```

```
         0         0   -1.1660

>> G=inv(DA)*R*U =

    1.0000   -0.1902   -0.6475
         0    1.0000    0.4918
         0         0    1.0000

>> A=Q*inv(R)*DA*G =

    0.7810   -0.0881   -0.6182
    0.3643   -0.5434   -0.7563
    0.0222   -0.0693   -0.9974

>> Rxx-A*A' =  1.0e-015 *

   -0.6661   -0.4441   -0.2220
   -0.4441   -0.4441   -0.1110
   -0.2220   -0.1110         0
(note in this case the null space [of H] component of Rxx is zero).

>> S0=DA*inv(D)*DA =

    1.9406    0.0000    0.0000
    0.0000    1.2929   -0.0000
    0.0000   -0.0000    2.9083

>> W=inv(S0-eye(3))*DA =

    1.0311    0.0000    0.0000
   -0.0000   -1.7299   -0.0000
   -0.0000   -0.0000   -0.6110

>> Gunb=eye(3)+S0*inv(S0-eye(3))*(G-eye(3)) =

    1.0000   -0.3924   -1.3358
   -0.0000    1.0000    2.1710
   -0.0000    0.0000    1.0000
```

Theorem 14.2.1 (Worst-case Noise GDFE as Best BC Receiver) *The single-user GDFE, with at least one appropriate single-user input, designed for a* **BC***'s worst-case noise achieves the highest possible (sum) data rate for the* **BC** *, which is* I_{wcn}*. Furthermore, the feedforward section of the GDFE is (block) diagonal and requires thus no coordination on the* **BC***. Finally, this unbiased MMSE GDFE necessarily performs exactly the same as the ZF-GDFE for this channel.* **Proof:** First, the diagonal GDFE feedforward section corresponds to the worst-case-noise for any channel with the appropriate input illustrated by construction in the text and example preceding this theorem. The a data rate I_{wcn} corresponds to this diagonal-W GDFE and represents the maximum possible data rate for the worst-case noise on this channel, or equivalently the rate of the MMSE-GDFE for this noise, input, and channel viewed as a single user. Any other set of receivers for the **BC** would necessarily have to also be a linear block diagonal. In fact, the mutual information for any user now viewed as one (possibly set) of the dimensions of the single-user input, with preceding Gaussian users added to the channel noise Gaussian noise, $I(\boldsymbol{x}_u; \boldsymbol{y}_u/[\boldsymbol{x}_1 \ldots \boldsymbol{x}_{u-1}]$, cannot be improved and is achieved by any (block) "scalar" multiplication and a subsequent maximum-likelihood (or mod-Λ_u) decoder. However, all these settings were considered as possible in the MMSE optimization of the GDFE, and had they lead to a higher sum rate than I_{wcn}, then they would have been selected by the MMSE-GDFE. Thus, this sum rate I_{wcn}cannot be exceeded. Lastly, because the individual SNR's (generally $\frac{|R_{\boldsymbol{xx}}|}{|\tilde{R}_{noise}(u)|}$) cannot be improved without co-operation among the different users' receivers, any scaling (multiplication by block-diagonal elements in general case) of the individual user outputs of the **BC** cannot change the set of SNRs nor the overall (sum) data rate. Thus, the feedforward section of the overall GDFE for this worst-case noise cannot improve the SNR's – it too, then, is useless in terms of overall data rate improvement. A ZF-GDFE for this same channel, noise, and input would be designed by QR factorization of $R_{wcn}^{-1}H$ also, and the multiplication by any diagonal or input-only-dependent upper triangular matrix as in step 7 of the preceding procedure for GDFE construction will remain zero forcing, and cannot change the overall performance. Indeed, the ZF-GDFE for the worst-case-noise-equivalent channel and the same input as the MMSE-GDFE could have been initially designed (although without the GDFE theory, the exact feedforward-diagonalizing decomposition of $R_{\boldsymbol{xx}}$ would have been obscured). **QED.**

This theorem's last part is ultimately the most useful, zero-forcing is the same as MMSE under worst-case noise (and this worst-case corresponds to best performance of the **BC** for any given input). Section 14.3 will introduce a simple dual GDFE construction and the ZF-GDFE will be obvious in that construction, and because of Theorem 14.2.1, it will be then known to be optimum.

The determination of the diagonalized GDFE receiver can be somewhat simplified when $\rho(H) < NL_yU$, in other words the channel is singular as is for instance the case on a simple $U \times 1$ channel where the rank is 1 and the **BC** channel matrix is tall. In this "degraded broadcast" case, the worst-case-noise design process follows this equation from Chapter 5

$$R_{wcn}^{-1}HAG^{-1}S_0^{-1}G^{-*}A^*H^*R_{wcn}^{-1} = D \tag{14.14}$$

where D is the same diagonal matrix that characterizes the original worst-case noise equation in (14.12). In this case, since H is singular, some of the diagonal terms of D will be equal to zero and only $\rho(H)$ of them will be nonzero. QR factorization of the matrix $R_{wcn}^{-1}H$ in this singular case causes a modification of the design procedure for the GDFE-diagonalizing input choice. For this singular case,

$$R_{wcn}^{-1}H = \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix} Q^* \quad . \tag{14.15}$$

The choice of A that solves (14.14) is

$$A = Q \begin{bmatrix} R^{-1} & 0 \\ 0 & I \end{bmatrix} D_A G \tag{14.16}$$

The matrix D_A is again diagonal. The follow steps then allow G and A to be computed:

1. compute R_{wcn} via [Rwcn,bwcn] = wcnoise(Rxx, H, Ly)

2. $D = R_{wcn}^{-1} - [HR_{\boldsymbol{xx}}H^* + R_{wcn}]^{-1}$, block diagonal $D = \begin{bmatrix} d & 0 \\ 0 & 0 \end{bmatrix} \geq 0$.

3. $\begin{bmatrix} R & 0 \\ 0 & I \end{bmatrix} Q^* = R_{wcn}^{-1}H$ via QR factorization. R is upper triangular of rank $\rho(H)$. Q is a (full-rank) orthogonal matrix.

4. $UU^* = Q^*R_{\boldsymbol{xx}}Q$, Cholesky factorization where U is upper triangular.

5. $D_A = \text{diag}\left\{\begin{bmatrix} R & 0 \\ 0 & I \end{bmatrix} U\right\}$.

6. $G = D_A^{-1}\begin{bmatrix} R & 0 \\ 0 & I \end{bmatrix} U$ (feedback section)

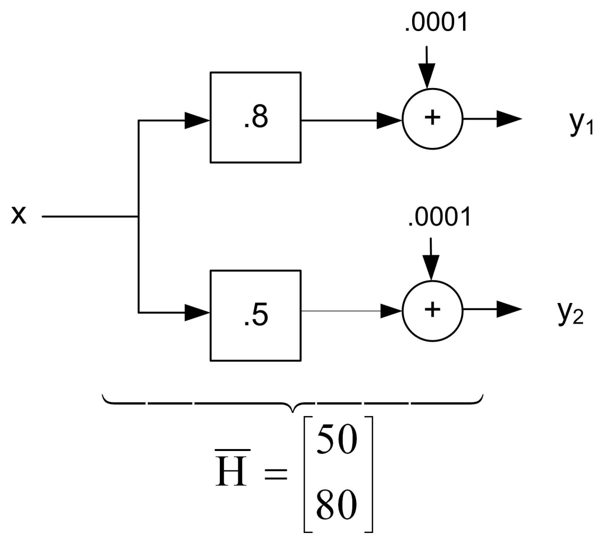


Figure 14.5: Simple BC example.

7. $A = Q \begin{bmatrix} R^{-1} & 0 \\ 0 & I \end{bmatrix} D_A G$ (transmit filter/matrix, so $R_{\boldsymbol{xx}} = AA^*$ is a check on this value of A).

8. $\begin{bmatrix} S_0^{-1} & 0 \\ 0 & 0 \end{bmatrix} = D_A^{-1} D D_A^{-1} = \begin{bmatrix} \boldsymbol{SNR} & 0 \\ 0 & 0 \end{bmatrix}$ (or simply $S_0^{-1}(i) = \frac{d(i)}{d_A^2(i)}$, $i = 1, ..., \rho(H)$.

9. $W^{unb} = \begin{bmatrix} (\boldsymbol{SNR} - I)^{-1} & 0 \\ 0 & 0 \end{bmatrix} D_A$ (feedforward filter).

10. $G^{unb} = I + \begin{bmatrix} \boldsymbol{SNR} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} (\boldsymbol{SNR} - I)^{-1} & 0 \\ 0 & 0 \end{bmatrix} (G - I)$ (unbiased feedback section for precoder)

The simple 1×2 BC channel of Section 12.3 is revisited here and repeated in Figure 14.5 for convenience. The total energy of the two user input energies, which are summed, is $\mathcal{E}_{\boldsymbol{x}} = 1$. The capacity region reappears in Figure 14.7. The capacity region for the channel was traced by using the precoder formulas with user 1 in the preferred position of order, producing the table:

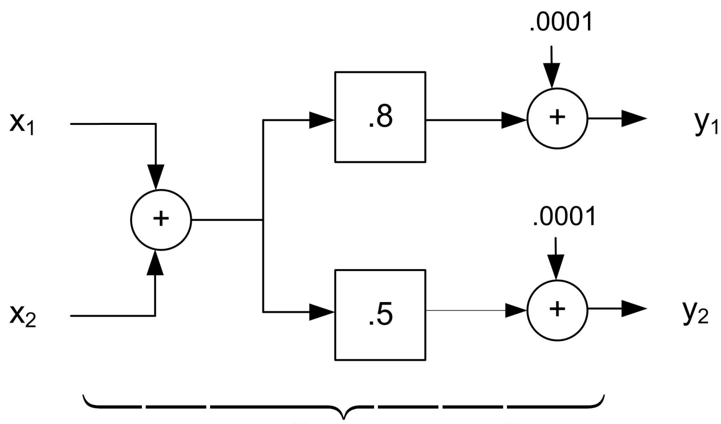
$\mathcal{E}_1$	$\mathcal{E}_2$	b_1	b_2	$b_1 + b_2$
1.0	0	6.32	0	6.32
.75	.25	6.12	.20	6.32
.50	.50	5.82	.50	6.32
.25	.75	5.32	1.0	6.32
.10	.9	4.66	1.66	6.32
.05	.95	4.16	2.20	6.26
0	1	0	5.64	5.64

Simple duality was used in Chapter 12 for the situation where each user on the BC used 1/2 unit of energy, and the corresponding MAC then had $\mathcal{E}_1 = 2501/2502$ and $\mathcal{E}_2 = 1/2502$, and the corresponding order was reversed (so the MAC had user 2 in the preferred position of going last in the MAC GDFE decoding order).

EXAMPLE 14.2.2 (return to simple 1×2 BC) For this BC, the worst-case noise can easily be found using software or simple calculus to be

$$R_{wcn} = \begin{bmatrix} 1 & \frac{5}{8} \\ \frac{5}{8} & 1 \end{bmatrix} \quad . \tag{14.17}$$

This R_{wnc} is only a function of total transmit energy and does not depend on the division of energy between the two users. A GDFE is somewhat trivial on this channel, since it



$$\overline{H} = \begin{bmatrix} 50 & 50 \\ 80 & 80 \end{bmatrix}$$

Figure 14.6: Equivalent Broadcast channel with summer viewed in channel.

estimates the input x. The mutual information corresponding to the worst-case noise is 6.322 bits/dimension, the maximum rate sum. Clearly, the GDFE would be diagonal with gain $W_1 = 1/80$ on user 1's receiver and $W_2 = 0$ on user 2's receiver because user 2's data rate at receiver 1 is always higher than user 2's data rate is at receiver 2. The mutual information for the two-dimensional channel-output vector $\boldsymbol{y}$, even though the GDFE feedforward section is diagonal and corresponding to worst-case noise, only measures the data rate that can be achieved by all users to all the receivers, including in particular receiver 1.

Thus, while worst-case noise and the GDFE are powerful concepts that simplify greatly multi-user channels, their use does yet truly consider each of the users' receivers in the BC. The clever reader may be tempted to construct the perfectly valid equivalent channel in Figure 14.6.

```
>>> H =
    80    80
    50    50

>> Rxx =
    0.5000         0
         0    0.5000

>> J2 =
     0     1
     1     0
>> [Rwcn,bsum]=wcnoise(Rxx,H,1)

Rwcn =
    1.0000    0.6250
    0.6250    1.0000

bsum = 6.3220

>> D=inv(Rwcn)-inv(H*Rxx*H'+Rwcn) =
```

```
    0.9998    0.0000
    0.0000    0.0000

>> Htilde=inv(Rwcn)*H =

   79.9999   79.9999
    0.0002    0.0002

>> [Q,R]=qr(J2*Htilde'*J2) ;
>> Q=J2*Q*J2;
>> R=(J2*R*J2)';
>> R=R*J2 =

 -113.1369   -0.0000
   -0.0002         0

>> Q=Q*J2 =

   -0.7071    0.7071
   -0.7071   -0.7071

>> R*Q'-Htilde = 1.0e-013 *

         0   -0.1421
         0         0
>> U=(J2*chol(J2*Rxxrot*J2)*J2)'=

    0.7071         0
         0    0.7071

>> r=R(1,1)=  -113.1369
>> Rnew=[r 0
0 1] =

 -113.1369         0
         0    1.0000

>> DA=diag(diag(Rnew*U)) =

  -79.9999         0
         0    0.7071

>> G=inv(DA)*Rnew*U =

     1     0
     0     1

>> A=Q*inv(Rnew)*DA*G =

   -0.5000    0.5000
   -0.5000   -0.5000

>> S0inv=inv(DA)*D*inv(DA);
```

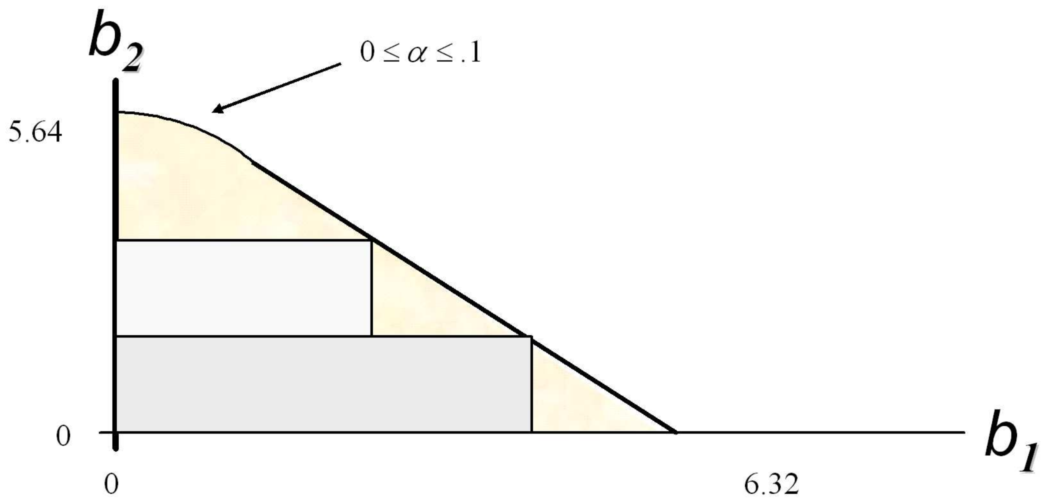


Figure 14.7: Rate region for broadcast channel

```
>> S0=[1/S0inv(1,1) 0
0 0] =

  1.0e+003 *

    6.4010         0
         0         0

>> Wunb= diag([ 1/(S0(1,1)-1) 0 ])*DA =

   -0.0125         0
         0         0
```

While the GDFE flows easily and estimates both inputs with worst-case noise as a diagonal feedforward filter, again user 2's path is zeroed because receiver 1 can always achieve a higher mutual information for any input energy distribution. The unasisted GDFE in this case is still attempting to estimate that maximum rate.

EXAMPLE 14.2.3 (simple BC via dual GDFE) For the same BC channel, this example instead notes that a GDFE MAC could be designed for the dual channel of Section 12.3. In particular, the channel used for duality was the transpose of the original BC, which then became a $U \times 1$ MAC. The GDFE designed for this MAC will provide a correct precoder and transmit filter (as well as diagonal feedforward) section for all the receivers' correct data rates for any energy distribution.

```
>> H=[50 80];
>> Rxx=[1/2502 0
0 2501/2502]  =

    0.0004         0
         0    0.9996
```

```
>> 0.5*log2(det(H*Rxx*H'+1))
    6.3219

>> Rbinv=H'*H+inv(Rxx) =  1.0e+003 *

    5.0020    4.0000
    4.0000    6.4010

>> Gbar=chol(Rbinv) =

   70.7248   56.5572
         0   56.5887

>> G=inv(diag(diag(Gbar)))*Gbar =

    1.0000    0.7997
         0    1.0000

>> S0=diag(diag(Gbar))*diag(diag(Gbar)) =  1.0e+003 *

    5.0020         0
         0    3.2023

>> b=.5*log2(det(Rxx*S0)) =
6.3219

>> b=.5*log2(diag(Rxx*S0)) =

    0.4997
    5.8222

>> SNR=Rxx*S0 =  1.0e+003 *

    0.0020         0
         0    3.2010

>> Wunb=SNR*inv(SNR-eye(2))*inv(S0)*inv(G')*H' =

    0.0200
    0.0125

>> Gunb=eye(2)+SNR*inv(SNR-eye(2))*(G-eye(2)) =

    1.0000    1.6000
         0    1.0000

>> Rvv=Rxx*inv(diag(Wunb)) =

    0.0200         0
         0   79.9680
```

Figure 14.8 illustrates the two GDFE's for the MAC and dual BC. The MAC design was easy and did not require knowledge of the worst-case noise. The feeback section has a feedback coefficient in both (the MAC GDFE feedback section and the BC GDFE precoder) that

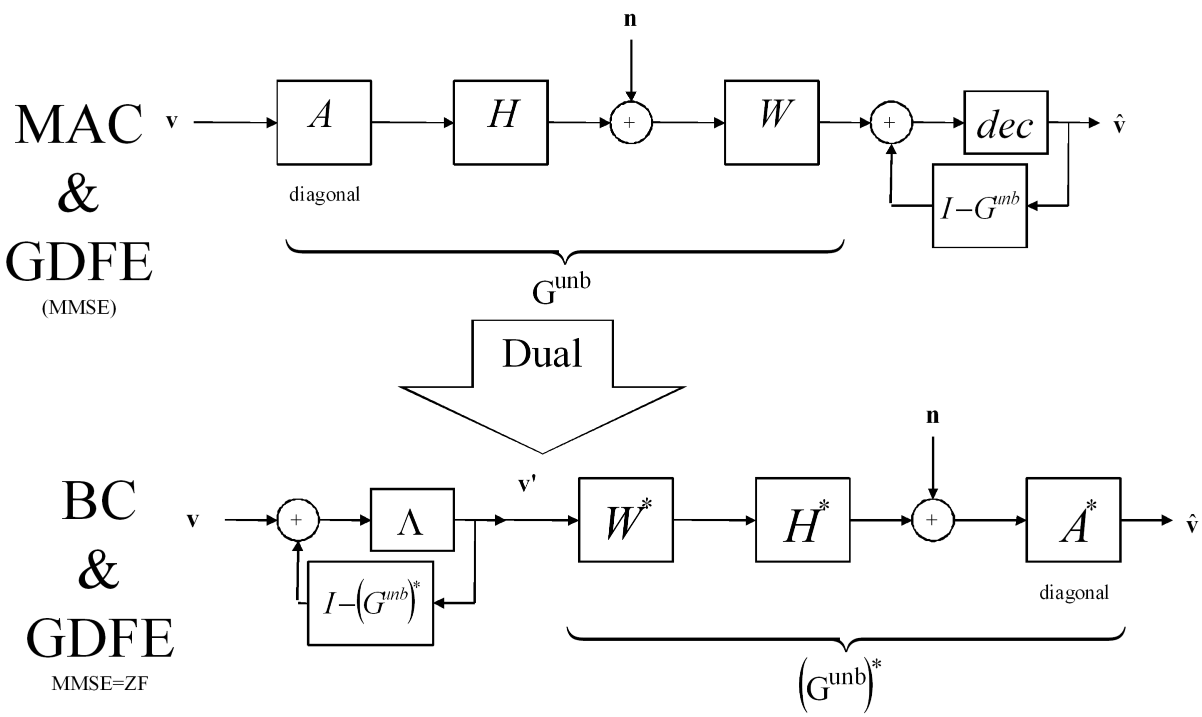


Figure 14.8: Dual GDFE.

is 5/8, which may have been obvious in this simple case but provides guidance to future designs with more complex channels. The diagonal (no coordination) transmit filters of the MAC correspond to the diagonal feedforward section of the BC channel GDFE. Similarly the transmit filter of the BC corresponds to the receive filter of the GDFE. In the case of the BC channel, the GDFE is both zero-forcing and MMSE (but only MMSE in the case of the MAC). This BC-only equivalence of ZF and MMSE makes clear that the simple transpose and reversal of the BC with respect to the MAC correspond to the correct optimum settings because they trivially synthesize the triangular feedback G^{unb}.

An issue is that the transmit constraint on energy must be satisfied correctly in each case, but Chapter 12 showed that a certain selection of the MAC energies will correspond exactly to the BC channel energies, and the bit rates for all users will be identical (in this example 1/2 and 1/2 on the BC channel energies while 2501/2502 and 1/2502 on the MAC). The energies for the transposed-reversed BC channel GDFE need simple adjustment on the precoder so that those going into the channel provide the correct bit rates. However, the realization of the transposed triangular channel is independent of these energies prior to the W matrix, and the cascade synthesizes the transpose of G^{unb}.

Those transmit energies are then

$$\mathcal{E}_{v,1} = (80)^2 \cdot \frac{1}{2} = 3200 \tag{14.18}$$

$$\mathcal{E}_{v,2} = (50)^2 \cdot \frac{1}{2} = 1250 \quad . \tag{14.19}$$

These energies are at both the input and output of the precoder, which on each of the two dimensions for the transmitted Guassian code words chooses the modulo for the infinite-dimensional hypersphere/Gaussian signal on each dimension to match these energies.

The example illustrates the key concept enabled by worst-case noise (which need not actually be computed in the dual formulation): the ZF-GDFE is the same as MMSE-GDFE on the worst-case-noise channel. This means that the synthesis (with a diagonal feedforward matrix) of a triangular $(G^{unb})^*$ that corresponds to the precoder is not only possible but optimum, and follows from the GDFE of the dual MAC.

14.2.2 Yu's Maximum Rate Sum for the BC

Former EE479 student Wei Yu was the first to find an expression for the maximum rate sum of the **BC**, which follows as that water-filling input $R^o_{\boldsymbol{xx}}$ for which the R_{wcn} equation is satisfied. Problem 14.9 further develops software for the simple convergent process of

1. Initialize $R_{\boldsymbol{xx}} = \frac{\mathcal{E}_{\boldsymbol{x}}}{N} \cdot I$.
2. Compute the R_{wcn} for the given $R_{\boldsymbol{xx}}$.
3. Compute the water-filling $R_{\boldsymbol{xx}}$ for the given R_{wcn}
4. If W is block diagonal, stop. Otherwise, return to step 2.

The stopping criterion could also be that successively computed $R_{\boldsymbol{xx}}$ are nearly equal, or successively computed R_{wcn} are nearly equal. The convergence is assured in that step 3's corresponding GDFE rate sum always must be less than that same rate sum on the previous instance of step 3 (if not the GDFE W would have already been diagonalized).

14.3 Vector Duality for the BC

This section extends the concept of duality to vector channels. Some care is initially necessary in this extension to avoid singularity and to construct of an appropriate square channel matrix for any channel (square or non-square) without loss of information. The dual channel will simply be the conjugate transpose of the original channel. By appropriate mapping of input energies between broadcast users and a dual set of multiple-access users, it will be possible to set all user rates equal while maintaining also the same energy sum. Such duality then allows ready and easy description of a dual GDFE, from which the best BC design can be quickly derived. The input autocorrelation matrices for the **BC** are determined through calculation of the dual autocorrelation matrices for the dual **MAC** channel.

Vector duality follows the scalar concept, but requires some mathematical sophistication to handle the various matrix generalizations of the scalar dual concepts. Subsection 14.3.1 introduces some refined channel duality concepts, defining a dual channel to be simply the conjugate transpose of a certain noise-equivalent channel. Subsection 14.3.2 then proceeds to use the duality concept to determine a set of autocorrelation matrices for the **BC** and its dual **MAC**. The more general approach taken here easily shows that the mutual information of dual channels must be the same (so the rate sums are the same) and that the sum of the energies are also the same, eliminating the need for the algebraic proof in Chapter 12.

14.3.1 Channel Equivalences

Any vector Gaussian channel can be written as

$$\boldsymbol{y} = H\boldsymbol{x} + \boldsymbol{n} \quad , \tag{14.20}$$

where H is an $l_y \times l_x$ matrix. The initial application of duality in this section will be to a square channel matrix H. Fortunately, any non-square channel can be transformed into a square channel with the same mutual information by using dummy variables. When $l_y = l_x$, the channel is square. Otherwise:

When $l_x > l_y$, the matrix H is augmented by $l_x - l_y$ new zeroed rows of l_x zeros each, at the bottom of the matrix:

$$H \rightarrow \begin{bmatrix} H \\ \mathbf{0} \end{bmatrix} \quad . \tag{14.21}$$

Similarly the noise $\boldsymbol{n}$, and thus channel output $\boldsymbol{y}$ can be extended by $l_x - l_y$ arbitrary dummy positions that are ignored in actual implementation and do not exist

$$\boldsymbol{y} \rightarrow \begin{bmatrix} \boldsymbol{y} \\ \text{don't care} \end{bmatrix} \quad ; \quad \boldsymbol{n} \rightarrow \begin{bmatrix} \boldsymbol{n} \\ \text{don't care} \end{bmatrix} \quad . \tag{14.22}$$

It is convenient to let the "don't care" noise be Gaussian and independent of all other dimensions. Such noise can have unit variance on each dimension so that

$$R_{\boldsymbol{nn}} \rightarrow \begin{bmatrix} R_{\boldsymbol{nn}} & 0 \\ 0 & I \end{bmatrix} \quad . \tag{14.23}$$

The noise equivalent channel is then

$$\bar{H} \rightarrow \begin{bmatrix} R_{\boldsymbol{nn}}^{-1/2} H \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} R_{\boldsymbol{nn}} & 0 \\ 0 & I \end{bmatrix}^{-1/2} \cdot \begin{bmatrix} H \\ \mathbf{0} \end{bmatrix} \quad . \tag{14.24}$$

The resultant noise-equivalent channel matrix $\bar{H}$ is square $l_x \times l_x$. The mutual information for the system remains $I(\boldsymbol{x}; \boldsymbol{y})$ because the extra zeroed rows do not change the information transfer.

When $l_y > l_x$, some dummy artificial input dimensions are created to make H square $l_y \times l_y$. The new channel matrix becomes

$$H \rightarrow [H \; 0] \quad , \tag{14.25}$$

with $l_y - l_x$ extra columns of l_y zeros each. The output $\boldsymbol{y}$ and noise $\boldsymbol{n}$ remain $l_y \times 1$ vectors. The input $\boldsymbol{x}$ has an extra $l_y - l_x$ zero elements at the bottom (these could be anything because of the zeros in the

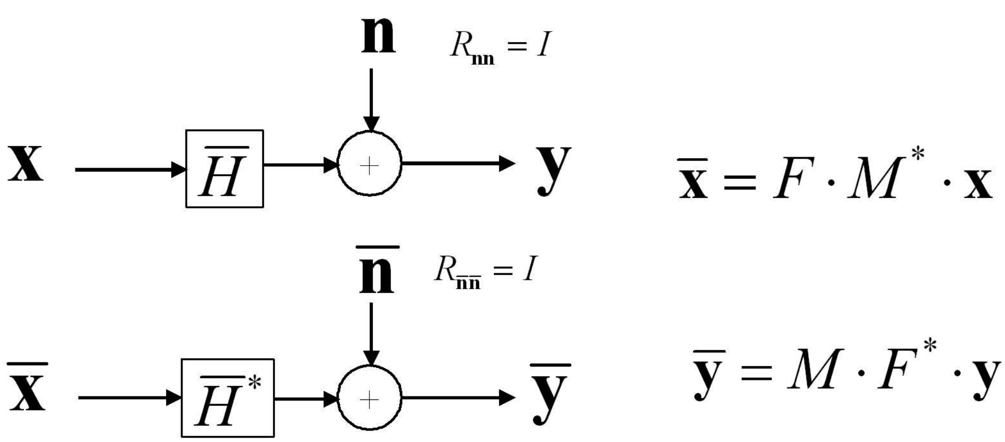


Figure 14.9: Illustration of the dual channel with white noise.

channel, but a choice of zero keeps energy to a minimum on the extra dummy dimensions)[5]. The input autocorrelation matrix becomes

$$R_{\boldsymbol{xx}} \rightarrow \begin{bmatrix} R_{\boldsymbol{xx}} & 0 \\ 0 & 0 \end{bmatrix} \quad . \tag{14.26}$$

The noise-equivalent channel is

$$\bar{H} = R_{\boldsymbol{nn}}^{-1/2} \cdot H \quad . \tag{14.27}$$

The mutual information remains $I(\boldsymbol{x};\boldsymbol{y})$.

For all channels the mutual information can always be computed according to (real baseband, remove 1/2 for complex case)

$$I(\boldsymbol{x};\boldsymbol{y}) = \frac{1}{2}\log_2 |\, I + \bar{H} \cdot R_{\boldsymbol{xx}} \cdot \bar{H}^* \,| \quad . \tag{14.28}$$

Theorem 14.3.1 (Dual Channels) *The square channel with white noise has a dual that is shown in Figure 14.9 both these channels have the same mutual information*

$$I(\boldsymbol{x};\boldsymbol{y}) = I(\bar{\boldsymbol{x}};\bar{\boldsymbol{y}}) \tag{14.29}$$

and the same input energy

$$trace\,(R_{\boldsymbol{xx}}) = trace\,(R_{\bar{\boldsymbol{x}}\bar{\boldsymbol{x}}}) \quad . \tag{14.30}$$

proof: The invariance of the mutual information follows from singular value decomposition of the square channel's noise-equivalent channel matrix

$$\bar{H} = F \cdot \Lambda \cdot M^* \quad . \tag{14.31}$$

Then

$$\begin{aligned}
\boldsymbol{y} &= F \cdot \Lambda M^* \cdot \boldsymbol{x} + \boldsymbol{n} && (14.32)\\
\boldsymbol{y}' \stackrel{\Delta}{=} F^*\boldsymbol{y} &= \Lambda \cdot (M^* \cdot \boldsymbol{x}) + \boldsymbol{n}' && (14.33)\\
&= \Lambda \cdot F^* \cdot (F \cdot M^* \cdot \boldsymbol{x}) + \boldsymbol{n}' && (14.34)\\
&= \Lambda \cdot F^* \cdot \bar{\boldsymbol{x}} + \boldsymbol{n}' && (14.35)\\
\bar{\boldsymbol{y}} \stackrel{\Delta}{=} M\boldsymbol{y}' &= M \cdot \Lambda \cdot F^* \cdot \bar{\boldsymbol{x}} + \bar{\boldsymbol{n}} && (14.36)\\
\bar{\boldsymbol{y}} &= \bar{H}^* \cdot \bar{\boldsymbol{x}} + \bar{\boldsymbol{n}} && (14.37)
\end{aligned}$$

[5] Calculations for normalized quantities to the number of dimensions should continue to use l_x and not l_y.

The noise $\bar{\boldsymbol{n}}$ also has an indentity for a covariance (since the original noise had an identity autocorrelation on the noise-equivalent channel and the new noise was obtained only through orthogonal transformations from original noise). Through the series of equations (14.32) - (14.37), all transformations were orthgonal and 1-to-1 so the mutual information is preserved. Equation (14.37) corresponds to the dual channel in Figure 14.9. Thus,

$$I(\boldsymbol{x};\boldsymbol{y}) = I(\bar{\boldsymbol{x}};\bar{\boldsymbol{y}}) \tag{14.38}$$

$$\frac{1}{2}\log_2 \mid I + \bar{H} R_{\boldsymbol{xx}} \bar{H}^* \mid = \frac{1}{2}\log_2 \mid I + \bar{H}^* R_{\bar{\boldsymbol{x}}\bar{\boldsymbol{x}}} \bar{H} \mid \quad . \tag{14.39}$$

Essentially then a square channel H with white noise, and its conjugate also with white noise, have the same mutual information when the inputs are related according to

$$\bar{\boldsymbol{x}} = F \cdot M^* \cdot \boldsymbol{x} \quad . \tag{14.40}$$

The input autocorrelation matrices are related by

$$R_{\bar{\boldsymbol{x}}\bar{\boldsymbol{x}}} = F \cdot M^* \cdot R_{\boldsymbol{xx}} \cdot M \cdot F^* \quad . \tag{14.41}$$

Because both of the transformations in (14.40), F and M^* are orthogonal, they are energy preserving and so the trace remains the same. **QED.**

When $l_x > l_y$, it is possible that the input $\boldsymbol{x}$ and consequently its autocorrelation matrix $R_{\boldsymbol{xx}}$ have energy in the null space of the channel. While this energy is useless, it created an initial need for the square channel. The energy of both the dual channel and the original channel

$$\mathcal{E}_{\boldsymbol{x}} = \mathcal{E}_{\boldsymbol{x}}(\text{pass}) + \mathcal{E}_{\boldsymbol{x}}(\text{null}) \tag{14.42}$$

because of our careful squaring of $\bar{H}$. If the rank of a non-square initial $\bar{H}$ is $\rho(\bar{H})$ when $l_x > l_y$, then the rank of a non-square initial $\bar{H}$ is $\rho(\bar{H})$ when $l_x > l_y$, then the F in the F (and M) in the singular value decomposition of the square channel may be replaced by the first $\rho(\bar{H})$ columns of F (M) (presuming the SVD associates these columns with the $\rho(\bar{H})$ non-zero singular values). In this case, then F and M will both become non square $\rho(\bar{H}) \times l_x$ matrices. Then,

$$F^* \bar{\boldsymbol{x}} = M^* \boldsymbol{x} \tag{14.43}$$

still holds. The quality on the right $M^*\boldsymbol{x}$ only contains components in the pass space of $\bar{H}$. Computing the squared norm of the vector written in (14.43)

$$\bar{\boldsymbol{x}}^* F F^* \bar{\boldsymbol{x}} = \boldsymbol{x}^* M M^* \boldsymbol{x} \tag{14.44}$$

$$\|\bar{\boldsymbol{x}}\|^2 = \|\boldsymbol{x}\|^2_{\text{pass}} \quad . \tag{14.45}$$

Thus, duality can be excuted with only the $\rho(\bar{H})$ columns of F and M in the equations above, and then energy in the pass space (which is all of concern for the channel) is the same. If the singular value decomposition of the original channel were instead executed directly, then the F obtained would have been directly an $l_y \times l_y$ matrix instead of the $\rho(\bar{H}) \times l_x$ matrix of columns formed an used in (14.43). The nonsingular columns of M remain the same and are still l_x dimensional in both singular value decompositions (of square $\bar{H}$ and non-square $\bar{H}$). However, (14.43) still holds, so the energies in the pass space remain equal. Clearly, the mutual information also remains equal between the dual and the original channel. Thus, the dual channel H^* may directly be formed without channel squaring at all as long as F and the M matrices are replaced by only the columns corresponding to non-zero singular values and Figure 14.9 holds without need of $\bar{H}$ being square. The dual channel and the original channel have the same mutual information and the same input energies.

The dual channels could be viewed with $\bar{H}$ as a **BC** and $\bar{H}^*$ as a MAC. Each user of the **BC** has a channel element $\bar{H}_u$ associated with it, and consequently the dual **MAC** user has a corresponding dual channel $\bar{H}_u^*$. This duality is at a user level, as well as the overall channels being duals from a single-user perspective. The overall system's energy equivalence tells us that the sum of the **BC** users' energy must

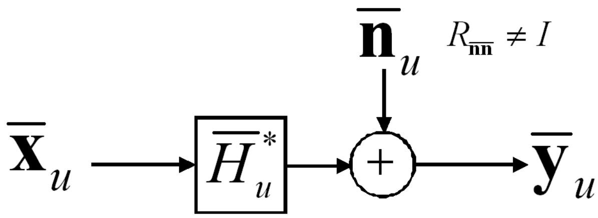


Figure 14.10: The transpose channel with different noise autocorrelation matrix.

equal the sum of the energies on the input to the **MAC** channel. However, the individual channels have white noise and corresponding bit rates b_u that are not the bit rates of interest in the **BC** and **MAC**. Those bit rates include "other-user" noise in calculations involving $\bar{H}_u$ and those other user noises are not yet included in the simple duality that occurs from the conjugate transpose. In fact, there is no guarantee that an input $R_{\boldsymbol{xx}}$ for the **BC** will correspond to an input with no coordination (that is a block-diagonal) autocorrelation matrix on the dual **MAC**. Further a certain set of autocorrelation matrices on the **MAC** channel need not necessarily correspond to a dual **BC** that would have a GDFE with diagonal feedforward section. The following **input deflection** concept provides the last missing element necessary for duality.

The following theorem is quite general, but will be used in the context of the deflecting matrix being the square root of a $\tilde{R}_{noise}(u)$ matrix for a given order in duality.

Theorem 14.3.2 (Input Deflection with Correlated Noise in Duality) *The two dual channels shown in Figure 14.11 have the same mutual information as the channel in Figure 14.10.* **Proof:** From Chapter 5, the invertible transformation of the channel input by matrix mutiplication does not change the mutual information. Or directly mathematically noting simply that there is a mutiplication by an identity does not change the expression,

$$\bar{H} \cdot R_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}}^{-1/2} \cdot \underbrace{\left(R_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}}^{1/2} \cdot R_{\boldsymbol{xx}} \cdot R_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}}^{*/2} \right)}_{R_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}}} \cdot R_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}}^{-*/2} \bar{H}^* = \bar{H} \cdot R_{\boldsymbol{xx}} \cdot \bar{H}^* \quad . \tag{14.46}$$

The mutual information remains the same[6], namely

$$I(\tilde{\boldsymbol{x}}; \boldsymbol{y}) = I(\boldsymbol{x}; \boldsymbol{y}) \quad , \tag{14.47}$$

when

$$\tilde{\boldsymbol{x}} = R_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}}^{1/2} \cdot \boldsymbol{x} \quad . \tag{14.48}$$

For this new channel at the top of Figure 14.11 between $\tilde{\boldsymbol{x}}$ and $\boldsymbol{y}$, the channel matrix is now

$$\tilde{\bar{H}} = \bar{H} \cdot R_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}}^{-1/2} \quad . \tag{14.49}$$

Thus deflection of the input can be viewed as corresponding to a similar deflection in the dual-channel noise-equivalent output. The dual for this new channel then derives from singular value decomposition on the new channel

$$\tilde{\bar{H}} = \tilde{F} \cdot \tilde{\Lambda} \cdot \tilde{M}^* \quad , \tag{14.50}$$

and leads to the lower channel in Figure 14.11 having the same mutual information as the original channel

$$I(\tilde{\tilde{\boldsymbol{x}}}; \tilde{\tilde{\boldsymbol{y}}}) = I(\tilde{\boldsymbol{x}}; \boldsymbol{y}) = I(\boldsymbol{x}; \boldsymbol{y}) \quad , \tag{14.51}$$

[6] The input energy, however, may not be maintained.

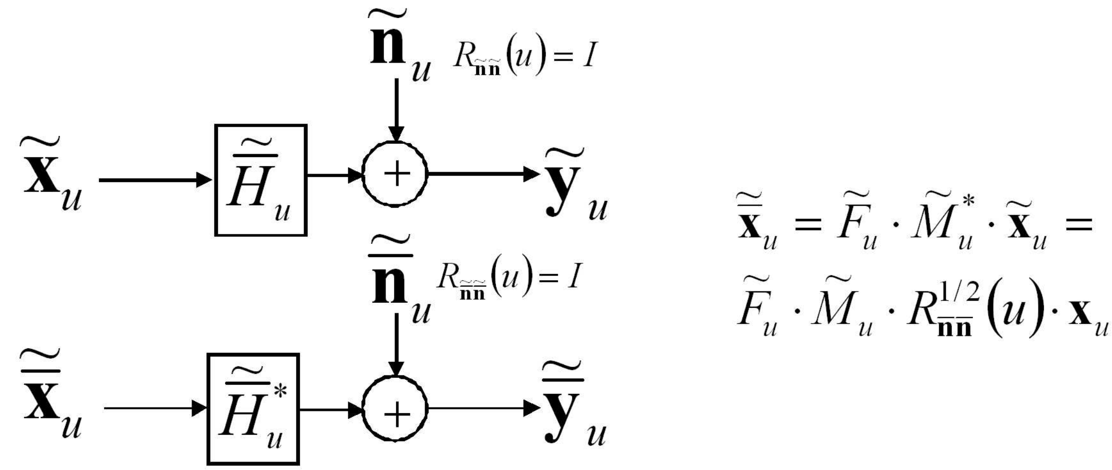


Figure 14.11: The dual of transpose channel with different noise autocorrelation matrix.

where the input transformation is

$$\tilde{\bar{\boldsymbol{x}}} = \tilde{F} \cdot \tilde{M}^* \cdot \tilde{\boldsymbol{x}} = \tilde{F} \cdot \tilde{M}^* \cdot R^{1/2}_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}} \cdot \boldsymbol{x} \quad . \tag{14.52}$$

The autocorrelation matrix relationships are

$$R_{\tilde{\bar{\boldsymbol{x}}}\tilde{\bar{\boldsymbol{x}}}} = \tilde{F} \cdot \tilde{M}^* \cdot R_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}} \cdot \tilde{M} \cdot \tilde{F}^* = \tilde{F} \cdot \tilde{M}^* \cdot R^{1/2}_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}} \cdot R_{\boldsymbol{x}\boldsymbol{x}} \cdot R^{*/2}_{\bar{\boldsymbol{n}}\bar{\boldsymbol{n}}} \cdot \tilde{M} \cdot \tilde{F}^* \quad . \tag{14.53}$$

This second noise-absorbing use of dual channels appears in the dual channel to absorb the user-dependent $\tilde{R}_{noise}$ terms that occur in the **BC** and **MAC**. **QED.**

This concept of input deflection and the dual channel will be applied not to the overall channel H 9where the overall duality preserved sum rate and total sum energy), but instead to the individual user data rates and channel elements in dual **BC** and **MAC** channels in Subsection 14.3.2.

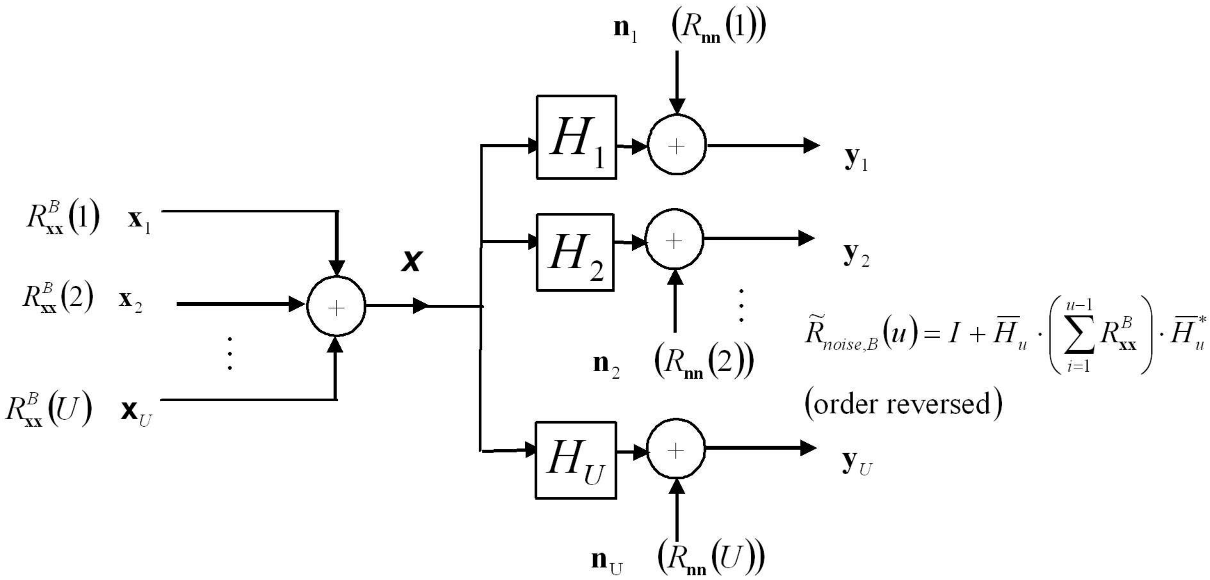
14.3.2 Duality

Figure 14.13 illustrates the **BC** and its **MAC** dual. The astute reader immediately notes that Theorem 14.3.2 applies directly to this channel (both noises are white) so that the rate sums of all users on both channels are equal and the total energy used by each channel is the same (that is the sum of the users energies is the same). However, duality can also be applied at an interior level, in particular with input deflection, to each of the user channels to cause the individual user rates to also be equal. This requires the use of input deflection on the individual channels (but the outer duality is also retained overall at all steps). Thus the individual user data rates will be set equal via input deflection for user-dependent colored noise, and so their energies will not be equal – however, the outer duality will be maintained always so that the rate sums and energy sums overall is maintained, as will follow:

The order on the **BC** is presumed to be reversed from the normal order with position U now as the least favorable and position 1 as the most favorable. The autocorrelation matrices will be indexed by a superscript of M for the **MAC** and B for the **BC**. Each system has an equivalent noise for each user that consists of other users not already cancelled or precoded:

$$\tilde{R}_{noise,M}(u) = I + \sum_{i=u+1}^{U} \bar{H}_i^* \cdot R^M_{\boldsymbol{x}\boldsymbol{x}}(i) \cdot \bar{H}_i \tag{14.54}$$

$$\tilde{R}_{noise,B}(u) = I + \bar{H}_u \cdot \left(\sum_{i=1}^{u-1} R^B_{\boldsymbol{x}\boldsymbol{x}}(i) \right) \cdot \bar{H}_u^* \quad . \tag{14.55}$$



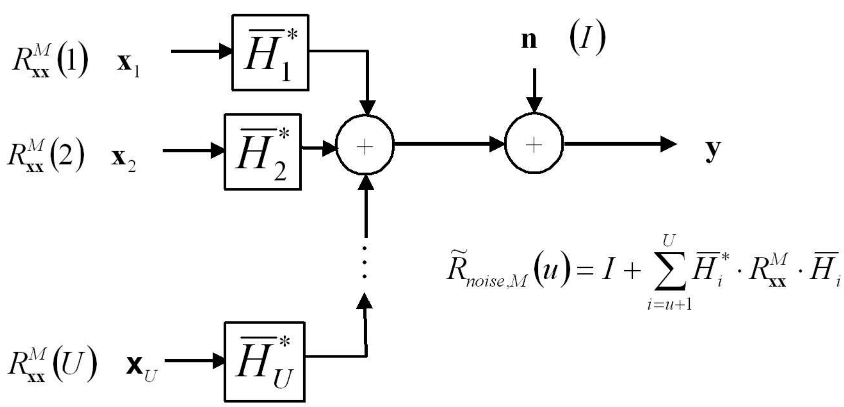


Figure 14.12: Dual Vector BC and MAC channels.

The data rates of each user on each of the two dual channels will be set equal, and those data rates are

$$b_u^{MAC} = \frac{1}{2}\log_2 \frac{|\ \bar{H}_u^* \cdot R^M_{\boldsymbol{xx}}(u) \cdot \bar{H}_u + \tilde{R}_{noise,M}(u)\ |}{|\ \tilde{R}_{noise,M}(u)\ |} \tag{14.56}$$

$$b_u^{BC} = \frac{1}{2}\log_2 \frac{|\ \bar{H}_u \cdot R^B_{\boldsymbol{xx}}(u) \cdot \bar{H}_u^* + \tilde{R}_{noise,B}(u)\ |}{|\ \tilde{R}_{noise,B}(u)\ |} \quad . \tag{14.57}$$

These user rates are the same as the original channels when viewed as **MAC** and **BC** respectively. These two individual user channels corresponding to the data rates in (14.56) and (14.57) look almost like duals except they have different noises. As in Figure 14.11, different noises can be accommodated by input deflections. Each individual channel can be viewed in terms of equivalent noise with crosstalk that is not white for its individual data rate. Thus each of these channels may have their input deflected by the square-root noise autocorrelation of the other, where the noise autocorrelation contains the other users as in Equations (14.54) and (14.55). Such input deflection does not change the mutual information, so the rates of the two channels with noise-input deflection are maintained equal, and thus (equating the arguments of the logarithms in (14.56) and (14.57))

$$\frac{|\ \bar{H}_u^* \cdot \tilde{R}^{-*/2}_{noise,B}(u) \cdot R^M_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}}(u) \cdot \tilde{R}^{-/2}_{noise,B}(u) \cdot \bar{H}_u + \tilde{R}_{noise,M}(u)\ |}{|\ \tilde{R}_{noise,M}(u)\ |} = \tag{14.58}$$

$$\frac{|\ \bar{H}_u \cdot \tilde{R}^{-/2}_{noise,M}(u) \cdot R^B_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}}(u) \cdot \tilde{R}^{-/2}_{noise,M}(u) \cdot \bar{H}_u^* + \tilde{R}_{noise,B}(u)\ |}{|\ \tilde{R}_{noise,B}(u)\ |} \quad , \tag{14.59}$$

where

$$\tilde{\boldsymbol{x}}_u^M = R^{*/2}_{noise,B}(u) \cdot \boldsymbol{x}_u^M \tag{14.60}$$

$$\tilde{\boldsymbol{x}}_u^B = R^{1/2}_{noise,M}(u) \cdot \boldsymbol{x}_u^B \quad . \tag{14.61}$$

By dividing the denominator noise into both terms, and defining

$$\tilde{\bar{H}}_u \triangleq \tilde{R}^{-1/2}_{noise,B}(u) \cdot \bar{H}_u \cdot \tilde{R}^{-/2}_{noise,M}(u) \quad , \tag{14.62}$$

the dual equations become (for setting each user's **MAC** and **BC** bit rates equal)

$$|\ \tilde{\bar{H}}_u^* \cdot R^M_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}}(u) \cdot \tilde{\bar{H}}_u + I\ |=|\ \tilde{\bar{H}}_u \cdot R^B_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}}(u) \cdot \tilde{\bar{H}}_u^* + I\ | \quad . \tag{14.63}$$

Using the results of Figure 14.11, then an SVD of $\tilde{\bar{H}}_u$ yields

$$\tilde{\bar{H}}_u = \tilde{F}_u \cdot \tilde{\Lambda}_u \tilde{M}_u^* \quad , \tag{14.64}$$

and a relationship of the two input autocorelation matrices as

$$R^M_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}}(u) = \tilde{F}_u \cdot \tilde{M}_u^* \cdot R^B_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}} \cdot \tilde{M}_u \cdot \tilde{F}_u^* \quad . \tag{14.65}$$

Furthermore, from the input deflections

$$R^M_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}}(u) = \tilde{R}^{*/2}_{noise,B}(u) \cdot R^M_{\boldsymbol{xx}}(u) \cdot \tilde{R}^{1/2}_{noise,B}(u) \quad , \tag{14.66}$$

and equivalently

$$R^B_{\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}}(u) = \tilde{R}^{1/2}_{noise,M}(u) \cdot R^B_{\boldsymbol{xx}}(u) \cdot \tilde{R}^{*/2}_{noise,M}(u) \quad . \tag{14.67}$$

Combing the equations (14.63), (14.65), and (14.67), the desired relation between **MAC** and **BC** covariances is the established as (BC to MAC)

$$R^M_{\boldsymbol{xx}}(u) = \tilde{R}^{-*/2}_{noise,B}(u) \cdot \tilde{F}_u \cdot \tilde{M}_u^* \cdot \tilde{R}^{1/2}_{noise,M}(u) \cdot R^B_{\boldsymbol{xx}}(u) \cdot \tilde{R}^{*/2}_{noise,M}(u) \cdot \tilde{M}_u \cdot \tilde{F}_u^* \cdot \tilde{R}^{-/2}_{noise,B}(u) \quad , \tag{14.68}$$

which reverses from simple algebra to the MAC-to-BC relationship

$$R^B_{\boldsymbol{xx}}(u) = \tilde{R}^{-/2}_{noise,M}(u) \cdot \tilde{M}_u \cdot \tilde{F}^*_u \cdot \tilde{R}^{*/2}_{noise,B}(u) \cdot R^M_{\boldsymbol{xx}}(u) \cdot \tilde{R}^{1/2}_{noise,B}(u) \cdot \tilde{F}_u \cdot \tilde{M}^*_u \cdot \tilde{R}^{-*/2}_{noise,M}(u) \quad . \quad (14.69)$$

The relationships are indexed by u. Recursive use of Equation 14.68 must start with user U and work down because each successive $\tilde{R}_{noise,M}(u)$ in (14.54) depends on higher indices of $R^M_{\boldsymbol{xx}}(i > u)$. Equation (14.69) must start with user 1 and work up because each successive $\tilde{R}_{noise,B}(u)$ in (14.55) depends on lower indices of $R^B_{\boldsymbol{xx}}(i < u)$.

The dual relationships correspond to a recursion that is illustrated in the flow chart of Figure **??**. In this case $\tilde{R}_{noise,B}(u)$ must be constructed recursively after each successive step of duality.

The overall channels (viewed from single-user perspective) are still duals and have the same rate sum. The energies of the individual users are not the same because of the input deflection that makes individual **BC** and **MAC** users' bit rates equal, or equivalently they were never the same in the original overall channel, just their sums were equal.

mac2BCMimo Program (Mohseni)

The mac2BCMimo program is provided for computing duals. The 3 inputs are:

1. $R_{\boldsymbol{xx}}(u)$ (called S by author of program), specified as U square $L_x \times L_x$ autocorrelation matrices (the program calls U instead K and L_x instead r). The user index is the last in this 3-dimensional tensor input.

2. $\bar{H}^*$, each of the user matrices specified in succession with the last index in this 3-dimensional tensor also corresponding to the user. The H in the program is actually the **MAC** channel not the **BC**.

3. $\boldsymbol{\pi}$ a $1 \times U$ order vector representing the order for the **MAC**.

```
There is only one ouput, which is a tensor of the \vbc autocorrelation matrices with again the user

% This function converts the covariace matrices in a Gaussian MAC to corresponding
% covariance matrices in its Dual BC. These covariance matrices achieve the
% same set of rates in the dual BC by the GDFE precoding coding scheme. The
% encoding order in the BC is the reverse of the decoding order in the MAC.
% The total number of users is denoted by K. The decoding order in the MAC
% is given by K by 1 vector pi. pi(k) is the user that is decoded kth in
% the successive decoding. H is a t by r by K matrix containing all the
% channel matrices of the MAC. H(:,:,k) is the channel matrix for user k in
% the MAC. t and r are the number of transmit and receive antennas in the
% BC. S is a r by r by K matrix containing the covariance matrices of the
% MAC. S(:,:,k) is the covariance matrix for user k in the MAC. G is the
% function output which will contain the covariance matrices of the BC.
% G(:,:,k) is the covariance matrix for user k. This function is for just
% an MIMO-BC and does not include parallel MIMO-BCs.

function G = mac2BcMimo(S, H, pi)


H = H(:,:,pi);
S = S(:,:,pi);


[t, r, K] = size(H);

Gtot = zeros(t,t);
```

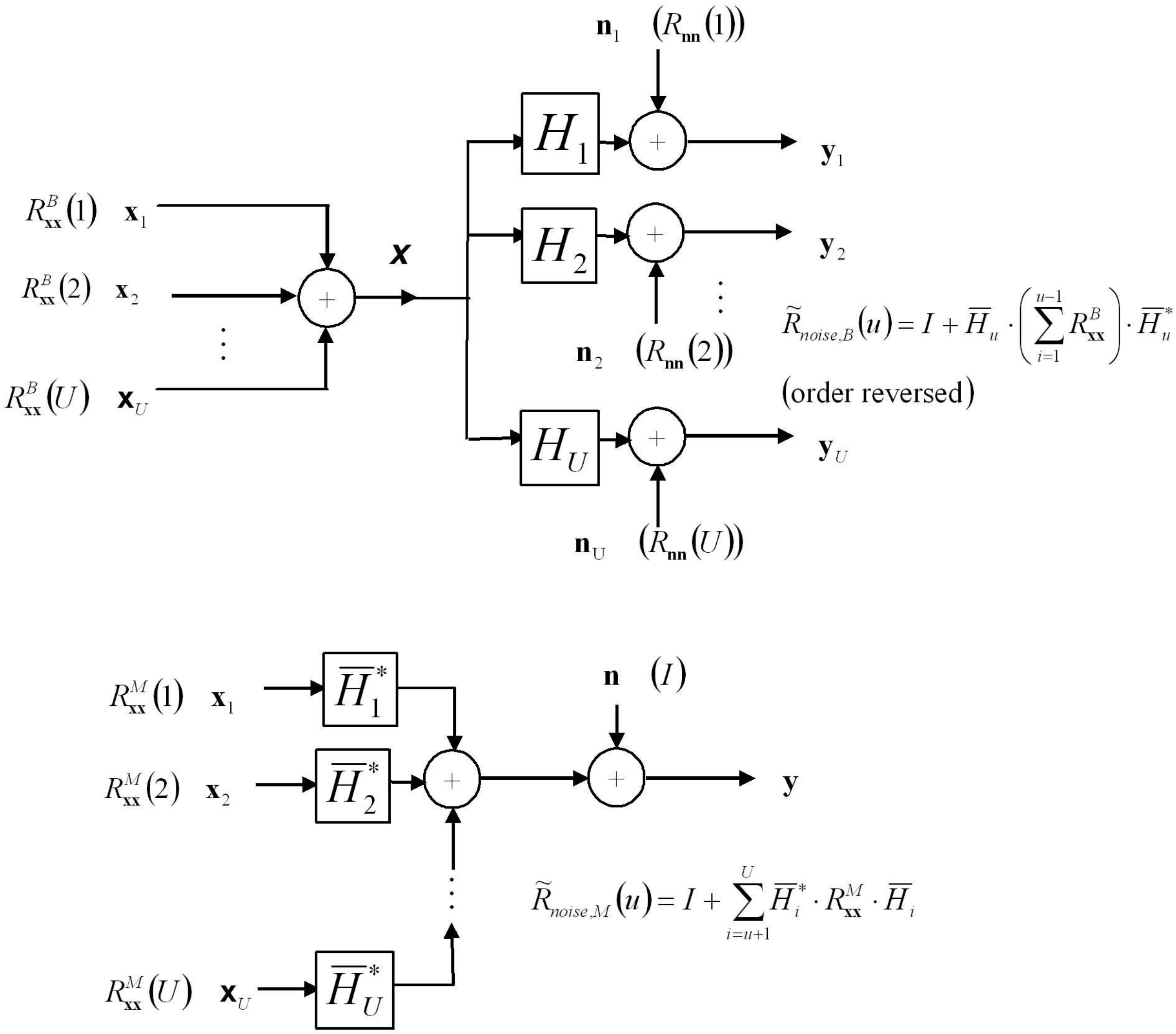


Figure 14.13: **BC** to **MAC** duality flow chart.

```
% Gtot is the transmit covariance matrix of the BC

B = zeros(t,t,K);
A = zeros(r,r,K);

% A and B matrices are the Rtildenoise for BC and MAC respectively


B(:,:,K) = eye(t);

for k = K:-1:2
   B(:,:,k-1) = B(:,:,k) + H(:,:,k)*S(:,:,k)*H(:,:,k)';
end

A(:,:,1) = eye(r);

for k = 1:K
   temp_A = inv(sqrtm(A(:,:,k)));
   temp_B = inv(sqrtm(B(:,:,k)));

   [F L M] = svd(temp_B * H(:,:,k) * temp_A);
   M = [M zeros(r,t-r)];
   G(:,:,k) = temp_B * F * M' * sqrtm(A(:,:,k)) * S(:,:,k) * sqrtm(A(:,:,k)) * M * F' * temp_B;

   Gtot = Gtot + G(:,:,k);

   if k~=K
      A(:,:,k+1) = eye(r) + H(:,:,k+1)' * Gtot * H(:,:,k+1);
   end
end

G(:,:,pi) = G;
```

14.3.3 Determination of $R_{\boldsymbol{xx}}$ for given rate tuple

The dual **MAC** for a **BC** allows ready determination of the feasibility of a rate tuple $\boldsymbol{b}$. This rate tuple can be inserted, along with $\bar{H}^*$, and the weight vector $\boldsymbol{w} = [111...1]$, as inputs to minPMAC. The resultant output energy should be summed and compared with the total energy constraint. If the total energy constraint is not exceeded, then the point is feasible (in the capacity region) and a GDFE can be designed for the set of user input energies on the dual **MAC**.

14.3.4 The dual GDFE

The dual GDFE for the set of input autocorrelation matrices $R_{\boldsymbol{xx}}(u)$ then can be translated as in Figure 14.8 of Section 14.2. The transmit autocorrelation matrices have to be ensured at the output of the transmitter A matrix, which means that each square term A_u must be prescaled by a matrix S_u so that $A_u S_u S_u^* A_u^* = R_{\boldsymbol{xx}}(u)$. The desired data rate vector $\boldsymbol{b}$ if feaasible on the dual **MAC** with energy less than the total constraint then corresponds exactly to the data rate vector on the **BC**.

EXAMPLE 14.3.1 (ISI and BC via duality) User 2 has a $1 + .9D$ channel with white input $\mathcal{E} = 1$ on the dual **MAC** channel. User 1 has a $1 - D$ channel with white input $calE_2 = 1$ on the dual **MAC** channel. Both **BC** output noises are white with variance $\sigma^2 = 0.181$:

```
>> P2=(1/sqrt(.181))*[1 .9 0
```

```
0 1 .9] =

    2.3505    2.1155         0
         0    2.3505    2.1155

>> P1=(1/sqrt(.181))*[1 -1 0
0 1 -1] =

    2.3505   -2.3505         0
         0    2.3505   -2.3505

>> H1=P1';
>> H2=P2';
>> H=[H2
H1]

H =

    2.3505         0
    2.1155    2.3505
         0    2.1155
    2.3505         0
   -2.3505    2.3505
         0   -2.3505

>> P=[P2 P1];
>> Rbinv=H*H'+eye(6) =

    6.5249    4.9724         0    5.5249   -5.5249         0
    4.9724   11.0000    4.9724    4.9724    0.5525   -5.5249
         0    4.9724    5.4751         0    4.9724   -4.9724
    5.5249    4.9724         0    6.5249   -5.5249         0
   -5.5249    0.5525    4.9724   -5.5249   12.0497   -5.5249
         0   -5.5249   -4.9724         0   -5.5249    6.5249

>> Gbar=chol(Rbinv) =

    2.5544    1.9466         0    2.1629   -2.1629         0
         0    2.6853    1.8517    0.2838    1.7737   -2.0575
         0         0    1.4305   -0.3674    1.1801   -0.8127
         0         0         0    1.2772   -0.7177    0.2234
         0         0         0         0    1.5225   -0.4967
         0         0         0         0         0    1.1553

>> G=inv(diag(diag(Gbar)))*Gbar =

    1.0000    0.7621         0    0.8467   -0.8467         0
         0    1.0000    0.6896    0.1057    0.6605   -0.7662
         0         0    1.0000   -0.2568    0.8249   -0.5681
         0         0         0    1.0000   -0.5619    0.1749
         0         0         0         0    1.0000   -0.3262
         0         0         0         0         0    1.0000

>> S0=diag(diag(Gbar))*diag(diag(Gbar)) =
```

```
    6.5249         0         0         0         0         0
         0    7.2107         0         0         0         0
         0         0    2.0463         0         0         0
         0         0         0    1.6312         0         0
         0         0         0         0    2.3182         0
         0         0         0         0         0    1.3346

>> b=0.5*log2(diag(S0)) =

    1.3530
    1.4251
    0.5165
    0.3530
    0.6065
    0.2082

>> Wunb=inv(S0-eye(6))*inv(G')*H =

    0.4254         0
    0.0522    0.3785
   -0.2137    0.4727
    0.4254   -0.1923
   -0.1814    0.2441
   -0.0107   -0.4254

>> Gunb=eye(6)+S0*inv(S0-eye(6))*(G-eye(6)) =

    1.0000    0.9000         0    1.0000   -1.0000         0
         0    1.0000    0.8006    0.1227    0.7669   -0.8896
         0         0    1.0000   -0.5023    1.6134   -1.1111
         0         0         0    1.0000   -1.4520    0.4520
         0         0         0         0    1.0000   -0.5737
         0         0         0         0         0    1.0000

>> Wbc=A' =

     1     0     0     0     0     0
     0     1     0     0     0     0
     0     0     1     0     0     0
     0     0     0     1     0     0
     0     0     0     0     1     0
     0     0     0     0     0     1

>> Abc=Wunb' =

    0.4254    0.0522   -0.2137    0.4254   -0.1814   -0.0107
         0    0.3785    0.4727   -0.1923    0.2441   -0.4254

>> Gbc=Gunb' =

    1.0000         0         0         0         0         0
    0.9000    1.0000         0         0         0         0
         0    0.8006    1.0000         0         0         0
```

```
   1.0000    0.1227   -0.5023    1.0000         0         0
  -1.0000    0.7669    1.6134   -1.4520    1.0000         0
        0   -0.8896   -1.1111    0.4520   -0.5737    1.0000

>> Rxxm1=eye(3);
>> Rxxm2=eye(3);
>> Rnoisem1=eye(2)+H2'*Rxxm1*H2 =

  11.0000    4.9724
   4.9724   11.0000

>> Rnoisem2=eye(2);
>> Rnoiseb1=eye(3);
>> Hbtilde1=inv(sqrtm(Rnoiseb1))*H1*inv(sqrtm(Rnoisem1)) =

   0.7728   -0.1846
  -0.9574    0.9574
   0.1846   -0.7728

>> [F1,L1,M1]=svd(Hbtilde1)

F1 =  -0.4082   -0.7071    0.5774
       0.8165         0    0.5774
      -0.4082    0.7071    0.5774

L1 =   1.6582         0
            0    0.5881
            0         0

M1 =  -0.7071   -0.7071
       0.7071   -0.7071
>> Rxxb1=inv(sqrtm(Rnoisem1))*M1*F1(1:3,1:2)'*inv(sqrtm(Rnoiseb1'))*Rxxm1*inv(sqrtm(Rnoiseb1))
*F1(1:3,1:2)*M1'*inv(sqrtm(Rnoisem1')) =

   0.1143   -0.0516
  -0.0516    0.1143


>> Rnoiseb2=eye(3)+H2*Rxxb1*H2' =

   1.6312    0.2828   -0.2568
   0.2828    1.6289    0.3370
  -0.2568    0.3370    1.5113

>> Hbtilde2=inv(sqrtm(Rnoiseb2))*H2*inv(sqrtm(Rnoisem2)) =

   1.7121   -0.0235
   1.5175    1.6910
  -0.0211    1.5409

>> [F2,L2,M2]=svd(Hbtilde2)

F2 = -0.4295    0.7412    0.5158
     -0.8161   -0.0741   -0.5731
```

```
     -0.3866   -0.6671    0.6368

L2 =  2.7799         0
           0    1.6557
         0         0
M2 = -0.7071    0.7071
     -0.7071   -0.7071
>> Rxxb2=inv(sqrtm(Rnoisem2))*M2*F2(1:3,1:2)'*inv(sqrtm(Rnoiseb2'))*Rxxm2*inv(sqrtm(Rnoiseb2))
*F2(1:3,1:2)*M2'*inv(sqrtm(Rnoisem2')) =

    0.5307   -0.0145
   -0.0145    0.5307

>> A1=Abc(:,4:6) =
    0.4254   -0.1814   -0.0107
   -0.1923    0.2441   -0.4254

>> A2=Abc(:,1:3) =
    0.4254    0.0522   -0.2137
         0    0.3785    0.4727
>> [Mb1,Sv1]=eig(Rxxb1)

Mb1 =
   -0.7071    0.7071
   -0.7071   -0.7071
Sv1 =
    0.0626         0
         0    0.1659
>> [Mb2,Sv2]=eig(Rxxb2)
Mb2 =
    0.7071    0.7071
   -0.7071    0.7071
Sv2 =
    0.5452         0
         0    0.5162

>> G12=G(4:6,1:3) =

    0.8467    0.1057   -0.2568
   -0.8467    0.6605    0.8249
         0   -0.7662   -0.5681

>> Feedback=Mb2*A2*G12*pinv(A2) =

    0.0173    0.1895
   -1.0634   -0.5055

(design separate GDFE for each user at receiver using Rxx(u), Hu, and earlier-user noise)
```

The example illustrates the appropriate setting of the $R_{\boldsymbol{xx}}(u)$ at the output of the transmit filter matrix A. This process basically aborbs the A_u into the feedback coefficients **between** different users. Since the point fed back is $\boldsymbol{x}_u$, then a pseudoinverse (inverts only on pass space and zeros in null space of A_u) ensures the correct input to the feedback portion of the precoder. The A_u has been "pushed" back through the modulo device and summer. The M_u matrix is a square root of $R^B_{\boldsymbol{xx}}(u)$ used to shape

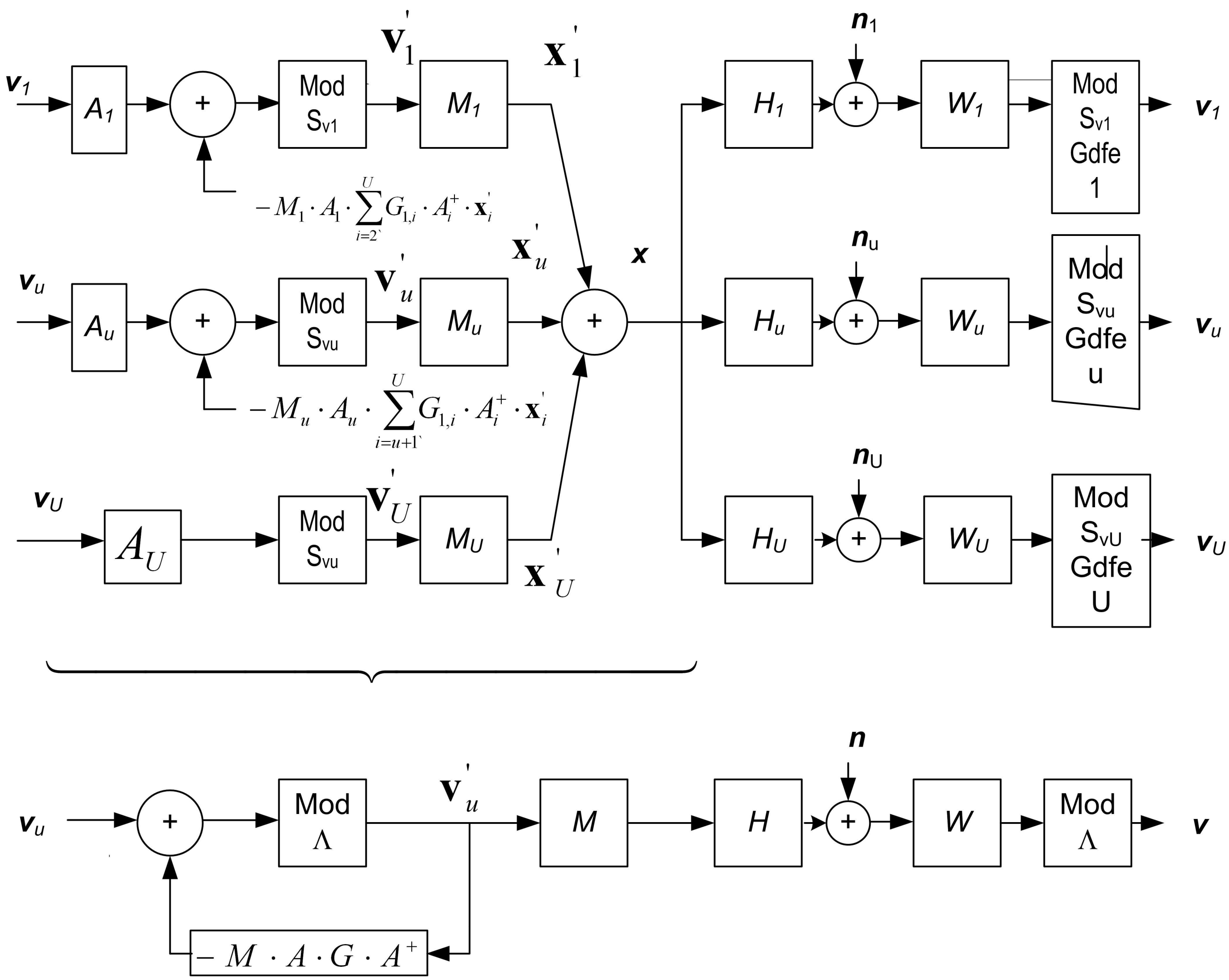


Figure 14.14: GDFE **BC** transmitter adjustment for matching dual energies.

the channel input, which also must then be used to filter the feedback subtractions since this alters the channel by this same amount.

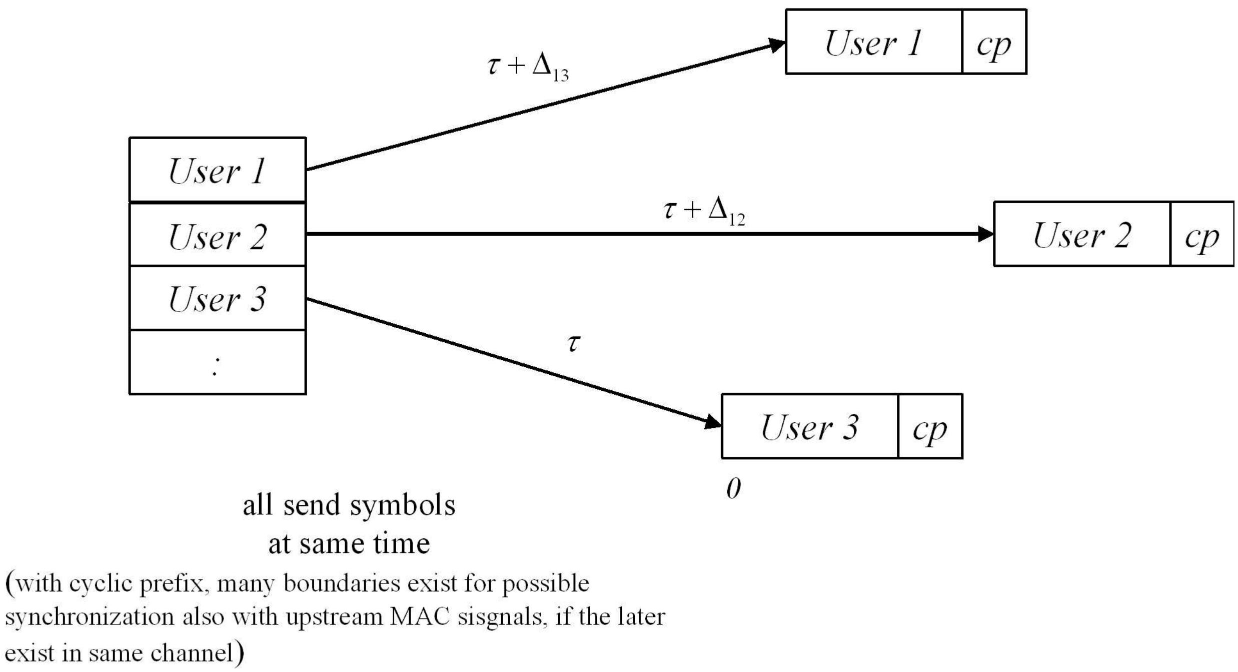


Figure 14.15:

14.4 Vector DMT and the BC

As with the **MAC**, vector DMT leads to enormous complexity reduction in stationary broadcast channels with intersymbol interference. Subsection 14.4.1 reviews VDMT and poses the alterations necessary for the **BC**. Subsection 14.4.1 will decompose (perhaps almost obviously at this point to the reader after Chapters 4, 5, and 13) into a set of many small parallel and largely independent scalar BCs on each tone. Finally, Subsection 14.4.2 concludes the VDMT section with an example in VDSL.

14.4.1 VDMT for the BC

Vector DMT systems were first introduced in Chapter 5 for the linear time-invariant channel and reviewed in Section 13.3 for the **MAC**. Section 4.8 also discusses the Zipper method by Isaksson that is presumed throughout here. The assumption of linear time invariance is continued throughout this section. Figure 14.15 illustrates the synchronization of the broadcast transmitter DMT symbols. Each transmitter uses the same size DMT symbol and aligns symbol boundaries so that all users arrive at a common receiver symbol boundary. The alignment is easier in broadcast channels than for the **MAC**. However, often systems are bi-directional, so both cyclic prefixes and cyclic suffixes can be used to align both directions of transmission for a downstream (or down-link) or an upstream (up-link) system so that all symbols at the common hub of multiple-access receivers and broadcast transmitters are aligned. Such alignment ensures that interference from "down back into up" can be easily cancelled at the hub with a single complex coefficient per tone. Such cancellers are often called "NEXT" cancelers (and are not the same as cancellers that exploit spatial correlation of noise in Section 13.3.4). Instead NEXT cancellers remove any effect from the broadcast signal into the recieved multiple-access signals.

Such alignment will, if the common cyclic extension of DMT partitioning is longer than the length of any of the response entries corresponding to each and all of the $\bar{H}_u$ (that is $\nu T' \geq$ length $\left\{\max_{u,i}\left(\tilde{\bar{h}}_{u,i}(t)\right)\right\}$, lead to no intersymbol interference and to crosstalk on any particular tone n that is a function ONLY of other users' signals on that same tone n of other users. Each tone of the $L_y U$ receivers' FFT outputs can then be modeled as

$$\underbrace{\boldsymbol{Y}_n}_{L_y U \times 1} = \underbrace{H_n}_{L_y U \times L_x} \cdot \underbrace{\boldsymbol{X}_n}_{L_x \times 1} + \underbrace{\boldsymbol{N}_n}_{L_y U \times 1} \quad , \tag{14.70}$$

where

$$H_n = \begin{bmatrix} H_{1,n} \\ \vdots \\ H_{U,n} \end{bmatrix} \tag{14.71}$$

$$\boldsymbol{X}_n = \begin{bmatrix} x_{1,n} \\ \vdots \\ x_{L_x,n} \end{bmatrix} \tag{14.72}$$

$$\boldsymbol{Y}_n = \begin{bmatrix} \boldsymbol{Y}_{1,n} \\ \vdots \\ \boldsymbol{Y}_{U,n} \end{bmatrix} \tag{14.73}$$

$$\boldsymbol{Y}_{u,n} = \begin{bmatrix} Y_{u,1,n} \\ \vdots \\ Y_{u,L_y,n} \end{bmatrix} \quad . \tag{14.74}$$

The $(l_y, l_x)^{th}$ entry of $H_{u,n}$ is the DFT of the response from line/antenna l_x to line/antenna l_y of user u's output. The energy constraint becomes

$$\sum_n \text{trace}\left\{R_{\boldsymbol{XX}}(n)\right\} \leq \mathcal{E}_{\boldsymbol{x}} \quad . \tag{14.75}$$

The input autocorrelation on tone n is

$$R_{\boldsymbol{XX}}(n) = \sum_{u=1}^{U} R_{\boldsymbol{XX}}(u,n) \quad . \tag{14.76}$$

This tone-indexed model for DMT leads to tremendous computational reduction with respect to the full precoding (or GDFE) structure. Effectively, N small channels of size $L_y \cdot U \times L_x$ replace a giant channel of size $L_y \cdot N \cdot U \times N \cdot L_x$. The GDFE/precoder computational advantage when $L_x = U$ and $L_y = 1$ is a complexity of $U \cdot N \cdot log_2(N) + NU^2$ versus the much larger $(N \cdot U)^2$, or if $N = 128$ and $U = 4$, the savings is a factor of about 50 times less computation (262,144 vs 5,632). Figure 14.16 is the result of the modeling.

The input for each tone then decomposes as in Section 14.2 or

$$R_{\boldsymbol{XX}}(n) = \sum_{u=1}^{U} R_{\boldsymbol{XX}}(u,n) = \sum_{u=1}^{U} A_{u,n} \cdot R_{\boldsymbol{V}'\boldsymbol{V}'}(u,n) \cdot A_{u,n}^* \quad . \tag{14.77}$$

A complete noise for any order (on any tone) can be found then as

$$\tilde{R}_{noise}(u,n) = R_{\boldsymbol{NN}}(u,n) + \sum_{i=u+1}^{U} H_{u,n} \cdot R_{\boldsymbol{XX}}(i,n) \cdot H_{u,n}^* \quad . \tag{14.78}$$

The noise-equivalent channel is then

$$\bar{H}_{u,n} = \underbrace{\tilde{R}_{noise}^{-1/2}(u,n)}_{L_y \times L_y} \cdot \underbrace{H_{u,n}}_{L_y \times L_x} \cdot \underbrace{A_{u,n}}_{L_x \times \rho_{u,n}} \quad . \tag{14.79}$$

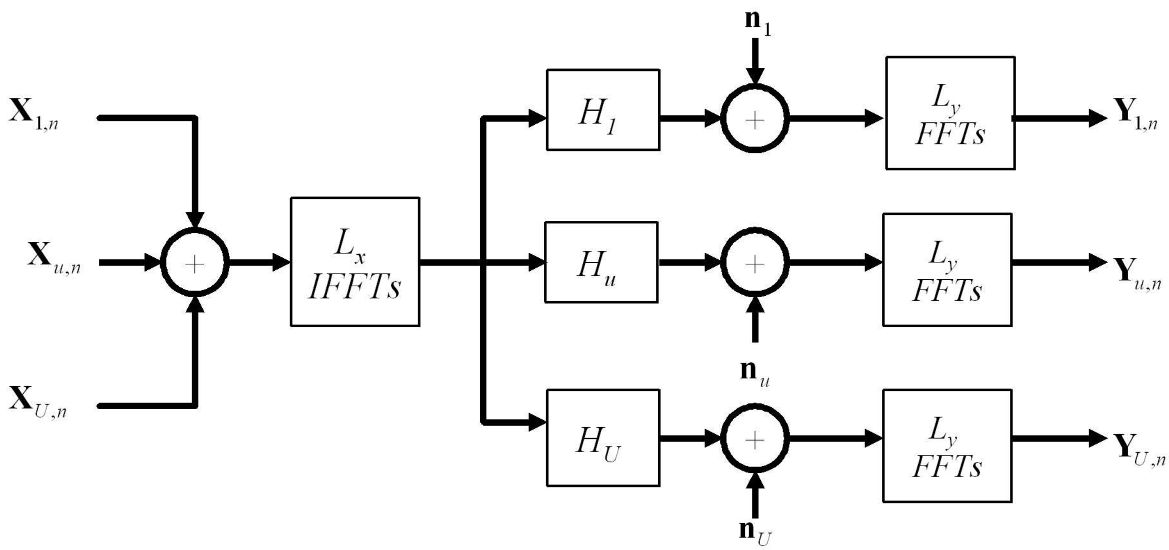


Figure 14.16: Illustration of the Vector DMT **BC** system.

Each tonal **BC** has a dual **MAC** and the overall dual of the VDMT **BC** is the VDMT **MAC** corresponding to the collection of the duals on each tone. Thus, for a rate vector $\boldsymbol{b}$ of all the users' desired rate, a minPMAC GDFE design can be checked for its total energy if less than the total allowed when the user-power weight vector $\boldsymbol{w}$ is uniform (all ones). The dual GDFE on each tone is then designed and translated into the **BC** GDFE with precoder on each tone. The input autocorrelation matrices for each of the **BC** users on each tone are computed by the duality procedure on each tone. Finally, the correct autocorrelation matrix on each tone for each user's component can be assured by a by

$$R_{\boldsymbol{xx}}(u,n) = A_{u,n} G_{p,u,n} R_{\boldsymbol{VV}}(u,n) G^*_{p,u,n} A^*_{u,n} \quad , \tag{14.80}$$

where $G_{p,u,n}$ is generalized monic triangular as in Section 14.3 and $R_{\boldsymbol{VV}}$ is a diagonal matrix of energies for each user.

14.4.2 Design Assessment

EXAMPLE 14.4.1 (VDSL) We visit again the vectored VDSL Example 13.3.1. In this case, the downstream direction is a **BC** if the DSLAM uses transmitter coordination as illustrated in Figure 14.17. This system then is vectored DMT if all downstream DMT transmissions use the same master clock. The tone spacing is 4.3125 kHz with a cyclic extension of 640 samples on a sampling clock of 16 × 2.208 MHz. Up to 8192 (VDSL2 so twice as many possible as in Example 13.3.1[7]) tones can be used in either direction. Three noise configurations have been suggested as of interest by all phone companies in North America. The first is a flat -125 dBm/Hz noise level (various levels of spatial correlation will be assumed for upstream, but note spatial correlation is of no consequence to downstream or more formally to the **BC**). This noise is considered to come from outside the coordinated lines and is not "analog front-end" noise. Configuration 1 attempts to model RF noise of radio signals for instance. Configuration 2 is a lower flat level of -140 dBm/Hz and is considered to be "analog front-end" noise and so would always have no spatial correlation (which again is only of interest for the upstream **MAC** that will also be shown in curves to follow). Configuration 3 is the same as Configuration 2 except that 6 T1 noise crosstalkers are assumed from outside the vectored group with varying degrees of spatial correlation.

[7] VDSL2 actually doubles carrier frequency width and leaves the number of tones at 4096 to cover exactly the same bandwidth, but the program used here was based on 4.3125 kHz spacing.

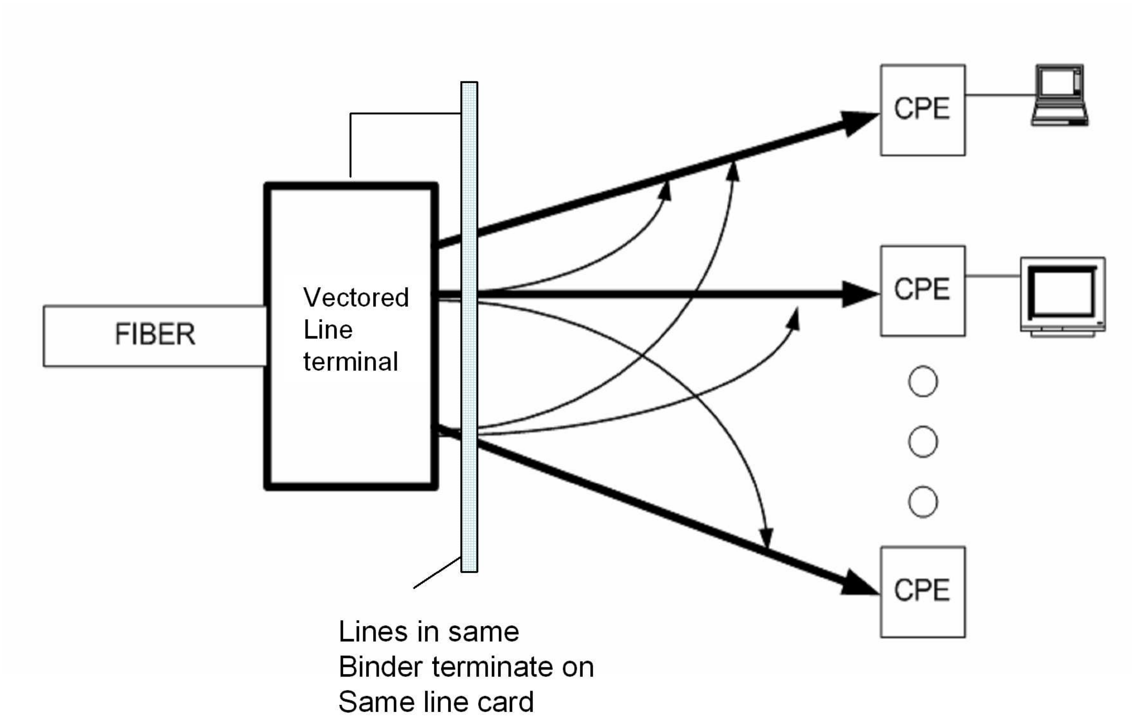


Figure 14.17: Downstream vectored BC for VDSL2.

Two frequency plans have been used for a frequency-division separation of upstream and downstream bands. The so-called 998 plan of North America allows up and down transmission below 138 kHz (tone 32), and also up-only transmission between 4 MHz and 5.2 MHz and and between 8.5 MHz and 17.6 MHz. These are the same as those in Example 13.3.1 were used except that now 7 bands are used with the highest additional cut-off frequencies set at 17 and 25 MHz. The results allow FDM of up and down so there is no up-into-down crosstalk of concern.

Figures 14.18 - 14.20 illustrate the achievable data rates with vectoring (up and down) with respect to goals posed by phone companies. As one can see, vectoring in either direction provides an enormous gain over expectations.

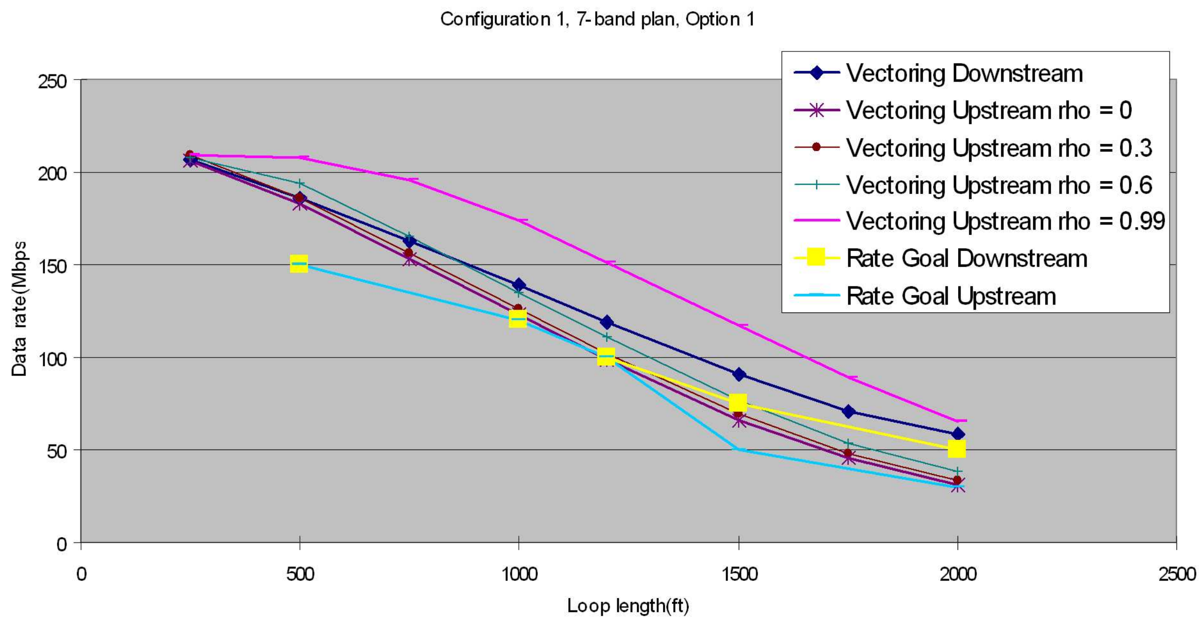


Figure 14.18: VDSL2 Data rates for **BC** and vdmt for Configuration 1.

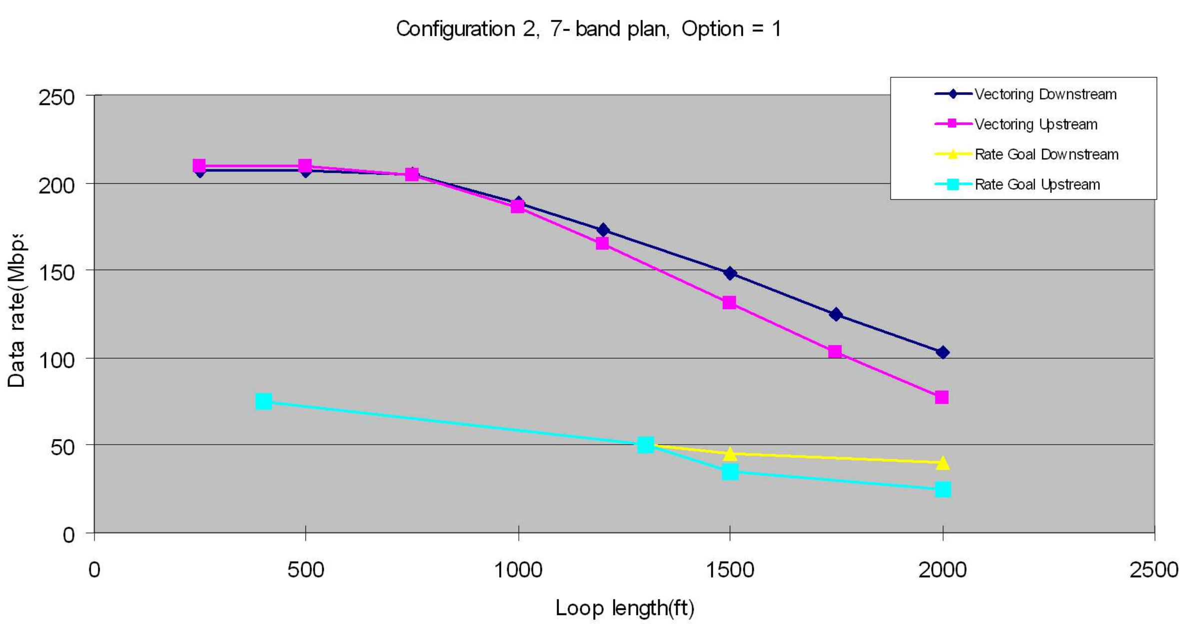


Figure 14.19: VDSL2 Data rates for **BC** and vdmt for Configuration 2.

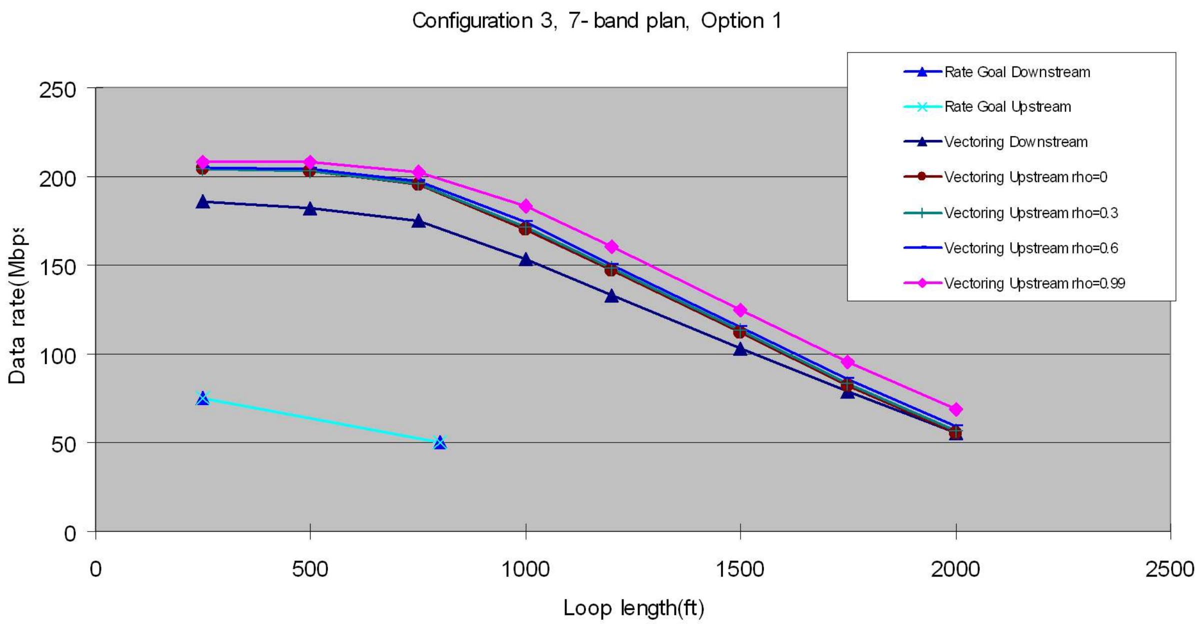


Figure 14.20: VDSL2 Data rates for **BC** and vdmt for Configuration 3.

14.5 Generation of the BC Capacity Rate Region

The steps for tracing the **BC** Capacity Region are:

1. create a dual **MAC** channel (with coefficients $\bar{H}^*$ and noise autocorrelation I.
2. for each $\boldsymbol{b}'$ with $b'_1 = 0, ..., b_{1,max}$, ... $b'_U = 0, ..., b_{U,max}$ with increments selected appropriately and maximums chosen sufficiently large to be outside the rate region (i.e., equal to the single user capacity for all other users zeroed)
   (a) Find the energy vector $\mathcal{E}_{vec}$ for a given $\boldsymbol{b}$ on the dual **MAC** using the minPmac program of Section 13.5.
   (b) if $\sum_u \mathcal{E}_u \leq \mathcal{E}_{vec}$, then the point is in the region, so $c_{new}(\boldsymbol{b}) = \{\boldsymbol{b}' \bigcup c_{old}(\boldsymbol{b})\}$.
3. Trace the boundary for all points in which $\sum_u \mathcal{E}_u = \mathcal{E}_{vec}$.

Exercises - Chapter 14

14.1 *Simple Gaussian Broadcast Channel.*
Given a1 $\times U$ Gaussian BC with

$$H = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_U \end{bmatrix} \text{ where } |h_1| > |H_i| \; \forall i > 1 \quad , \tag{14.81}$$

and $R_{\boldsymbol{nn}}(u) = 1 \; \forall u$, let $h_1 = 200$. The transmit symbol has energy 1 unit.

a. (2 pts) What is the maximum rate sum? (call this I_{wcn}. (2 pts)

b. (2 pts) Is the solution to part a "fair" to all users?

c. (2 pts) Is the mutual information with worst-case noise dependent upon the energy distribution among the users on this channel?

d. (2 pts) Can I_{wcn} be achieved by some GDFE receiver that has a diagonal feedforward matrix?

e. (2 pts) Does this I_{wcn} correspond to the maximum rate sum for all energy distributions that sum to the unit energy constraint?

14.2 *GDFE with WCN.*
Given a1 $\times U$ Gaussian BC with

$$H = \begin{bmatrix} .9 & 1 & 0 & 0 \\ 0 & .9 & 1 & 0 \\ 0 & 0 & .9 & 1 \end{bmatrix} \tag{14.82}$$

and $R_{\boldsymbol{xx}} = I$, while the channel output Gaussian noise has variance per dimension 0.181 on each output dimension.

a. (2 pts) Find R_{wcn} and b_{wcn} (2 pts)

b. (6 pts) Show a GDFE with a loss precoder at the xmit and a diagonal feedfoward matrix/filter realization.

14.3 *Dual Design for simple Gaussian BC.*
Given a1 $\times U$ Gaussian BC with

$$H = \begin{bmatrix} .8 \\ .5 \\ .3 \end{bmatrix} \tag{14.83}$$

and $R_{\boldsymbol{nn}}(u) = .0001 \; u = 1, 2, 3$. The transmit symbol has energy 1 unit.

a. (2 pts) Find R_{wcn} and b_{wcn}.

b. (2 pts) Find a GDFE realization for which W^{unb} is diagonal and G^{unb} is realized as a lossless precoder at the transmitter.

c. (4 pts) Suppose $\mathcal{E}_1$=.8, $\mathcal{E}_2$=.5, and $\mathcal{E}_3$=.3 , what is $\boldsymbol{b}$ =?

d. (4 pts) Show the dual MAC channel and its corresponding GDFE?

e. (4 pts) Show an acceptable GDFE realization for the original channel with the user data rates in part b?

14.4 *Zero Forcing Vector Code (8 pts).*

A **BC** has $L_y > 1$ and thus each user may have a multi-dimensional "vector" receiver.

a. (1 pt) In user u's receiver (for any u, does any use of a feedforward matrix for that user's L_y dimensions need to be diagonal and thus correspond to a worst-case noise for user u?

b. (2 pts) Following part a, can any linear matrix filter W_u change the data rate of user u? Why or why not?

c. (3 pts) From Section 14.2, WCN was observed to equate the performance of a ZF-GDFE and a MMSE-GDFE. Could a ZF-GDFE be used for user u alone? If your answer is yes, state the precise form of that ZF-GDFE.

d. (2 pts) If $H = QR$ is a qr factorization, is there a simple way to describe the action of the GDFE using Q and R on the broadcast channel without resorting to worst-case noise calculation (and without using duality)?

14.5 *Admission Problem (9 pts).*

For the BC $H = \begin{bmatrix} .9 \\ .6 \end{bmatrix}$ with $\sigma^2 = .01$, the input is restricted to unit energy per sample/symbol.

a. (2 pts) Sketch the rate region for this channel.

b. (2 pts) Using only the minPMAC software from Chapter 3, devise a scheme to dtermine if a specific rate vector $\boldsymbol{b} \in c(\boldsymbol{b})$.

c. (3 pts) Test your answer in part b for 3 largest points in capacity region that satisfy $b_2 = 10b_1$, $b_1 = 10b_2$, and $b_1 = b_2$.

d. (2 pts) What would it mean in the use of minPMAC for testing broadcast points if the weight vector were not uniform?

14.6 *Reciprocity (7 pts).*

A wireless hub system uses the same frequencies for uplink and downlink transmission (with a ping-pong or time-division access). The channel varies slowly. The noises at each of the downlink locations is the same variance and independent, as are the noises at the common uplink receiver. The wireless channel is said to exhibit reciprocity in this situation in that the downlink channel is the conjugate transpose of the uplink channel. All uses have symmetric transmission. The downlink total transmit power is restricted, but not the indivdual user's signals, and this total is the same as the sum of the uplink transmit powers.

a. (2 pts) Explain how a designer might avoid the need for feedback of channel information to the transmitters.

b. (2 pts) Is the treatment of the downlink and uplink transmitters in the way described ever optimal? If so, when?

c. (3 pts) Why might such a system not be used in practice? (give 3 reasons).

14.7 *Wifi.*

An 802.11(N) transmission system uses 64 tones in an OFDM/DMT arrangement with 4 transmit antennas at an access point and a cyclic prefix of 32 complex samples. Each of 3 receivers can use two receive antennas.

a. (3 pts) Determine L_x, U, and N for this application.

b. (4 pts) What are the dimensions of the matrices H_u and of the matrix H itself with and without the use of vector DMT?

c. (3 pts) For user 3, describe and draw a crypto precoder.

14.8 *DSL*

An VDSL transmission system uses 4096 tones in a baseband DMT arrangement with 25 twisted-pairs in the same crosstalking binder at a transmitter. The cyclic prefix contains 40(16)=650 real samples. Each of 25 separated receivers in different homes can receive downstream signals.

a. (3 pts) Determine L_x, U, and N for this application.

b. (4 pts) What are the dimensions of the matrices H_u and of the matrix H itself with and without the use of vector DMT?

c. (4 pts) For user 4, describe and draw a crypto precoder and describe the associated receiver.

14.9 *Rate Sum Maximization.*

This problem develops and tests software to determine the Broadcast Channel rate sum maximum. The specific example will be for $U = 3$, but the program should work for any number of users.

a. (3 pts) Find or write a water-filling program for a vector channel specified by H and an input total energy constraint trace$(R_{\boldsymbol{xx}}) \leq \mathcal{E}_{\boldsymbol{x}}$. The input can have up to N dimensions. Test this program on the channel specified in Section 14.2's example.

b. (4 pts) Write an program that iterates between the use of your water-filling program in part a and WCN determination for a set of noise variances. Use this on the same channel as part a to determine the noise and input that simulataneously satisfy water-filling and WCN.

c. (1 pts) Determine the Maximum rate sum.

d. (3 pts) Determine the diagonal GDFE matrix W that corresponds to your answer in part c.

e. (4 pts) Draw the transmitter and receiver for the maximum-rate-sum achieving system specified earlier in this problem.

f. (2 pts) Determine the data rates of each of the 3 users that achieve this maximum rate sum point.

14.10 *Sum capacity of MIMO Broadcast channels: (10 pts).*

Consider a vector broadcast channel with $U = 2$ users given by

$$\begin{bmatrix} \mathbf{y_1} \\ \mathbf{y_2} \end{bmatrix} = \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} \mathbf{x} + \begin{bmatrix} \boldsymbol{n}_1 \\ \boldsymbol{n}_2 \end{bmatrix}$$

where $\boldsymbol{y}_1, \boldsymbol{y}_2$ are $L_y \times 1$ received vectors, $\boldsymbol{x}$ is $L_x \times 1$ transmitted vector and $\boldsymbol{n}_1, \boldsymbol{n}_2 \sim \mathcal{N}(0, I)$ are receivers' noises. Let $\mathcal{E}_{\boldsymbol{x}}$ denote on the total transmit power. Based on the duality theory, the maximum rate-sum of this BC that is achievable by using "Dirty Paper Coding" scheme (or the lossless pre-coder studied in this course) is equal to maximum rate-sum of an equivalent MAC with channel matrices H_1^* and H_2^* and sum power constraint equal to $\mathcal{E}_{\boldsymbol{x}}$. This problem's objective is to prove this maximum rate-sum is actually equal to the sum-rate capacity. Let S_1 and S_2 denote on the transmit covariance matrices of user 1 and user 2 in the equivalent MAC respectively. As it was shown earlier, optimal S_1 and S_2 must simultaneously satisfy the water-filling conditions and since there is a total power constraint, the water level is the same for both users. Let $1/2\lambda$ be the optimal water-level, then the water-filling conditions can be written as,

$$\begin{aligned} \frac{1}{2} H_1 \left(H_1^T S_1 H_1 + H_2^T S_2 H_2 + I \right)^{-1} H_1^T + \Phi_1 &= \lambda I, \\ \frac{1}{2} H_2 \left(H_1^T S_1 H_1 + H_2^T S_2 H_2 + I \right)^{-1} H_2^T + \Phi_2 &= \lambda I, \end{aligned}$$

where Φ_1, Φ_2 are two symmetric positive semi-definite matrices satisfying $\mathrm{Tr}(S_1\Phi_1) = \mathrm{Tr}(S_2\Phi_2) = 0$. These conditions together with the power constraint $\mathrm{Tr}(S_1)+\mathrm{Tr}(S_2) = \mathcal{E}_{\boldsymbol{x}}$, are also known as the Karush Kuhn-Tucker optimality conditions and you do not need to verify them. Define the symmetric matrix Q to be

$$Q = \frac{1}{2\lambda}\left(H_1^T S_1 H_1 + H_2^T S_2 H_2 + I\right)^{-1}.$$

Consider a point to point MIMO channel

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{z}$$

with Gaussian noise vector $\boldsymbol{z} \sim \mathcal{N}(0, Q)$ and the same transmit power constraint $\mathcal{E}_{\boldsymbol{x}}$.

a. Justify the claim that the capacity of this point to point channel is an upper bound on the sum-capacity of the original BC. For this part you cannot use the fact that sum-capacity of the original BC is actually equal to the maximum rate-sum obtained by dirty paper coding scheme.

b. Using the KKT conditions given above, prove that the optimal covariance matrix that achieves the capacity of the point-to-point channel is equal to

$$R_{\boldsymbol{xx}} = \frac{1}{2\lambda}I - Q.$$

In other words, prove that $R_{\boldsymbol{xx}}$ is positive semi-definite, has trace less than or equal to $\mathcal{E}_{\boldsymbol{x}}$ and satisfy the water-filling condition.

c. Show that the capacity of this point to point channel that is obtained for the transmit covariance matrix of part (b) is equal to maximum rate-sum of the equivalent MAC and hence it is equal to maximum rate-sum of the original BC. Use this equality together with the upper bound property of part (a) to conclude that the maximum rate-sum of dirty paper coding scheme is equal to sum capacity and both are equal to capacity of the point to point channel.

14.11 *BC capacity region in a two-user-two-tone case (10 pts).*
This problem tries to find the BC capacity region boundary in a 2-user-2-tone case.

a. (2 user single tone case) Start with a simple single-tone case and find the capacity region of the following BC. $\bar{\mathcal{E}}_x = 1$ and $\sigma_n^2 = 1$ for both users. (3 pts)

$$y_1 = 2 \cdot x + n_1 \tag{14.84}$$

$$y_2 = x + n_2 \tag{14.85}$$

Hint. Placement of the user with a better channel in a preferable decoding position traces the region boundary. Subsequent division and assignment of the total power to each user appropriately determines each user's rate.

b. (Geometric Programming) Since the capacity region is convex, its boundary can be found by maximizing the linear combination of each user's rate, i.e.,

$$\begin{array}{ll} \max & a_1 b_1 + a_2 b_2 \\ \text{s.t} & b_1 \le \frac{1}{2}\log_2(1 + 2^2\mathcal{E}_1) \\ & b_2 \le \frac{1}{2}\log_2\{1 + \frac{\mathcal{E}_2}{1+\mathcal{E}_1}\} \\ & \mathcal{E}_1 + \mathcal{E}_2 \le 1, \end{array}$$

where $a_1, a_2 \in [0, 1]$ and $a_1 + a_2 = 1$. Show that this problem can be reformulated into the following geometric programming form. Geometric programming can be easily converted into a convex form and be efficiently solved by the existing convex optimization tool.

$$\max \quad f_0(x) \tag{14.86}$$

$$\text{s.t} \quad f_i(x) \le 1, \quad i = 1, \cdots, m, \tag{14.87}$$

where the constraint $x \succ 0$ is implicit and $f_0, \cdots, f_m$ are in the form of $f(x) = \sum_{k=1}^{K} c_k x_1^{a_1 k} x_2^{a_2 k} \cdots x_n^{a_n k} \quad (c_k > 0)$. This kind of function is called a *posynomial function.* (3 pts)

Hint. Remove $\mathcal{E}_1$ and $\mathcal{E}_2$ terms using the total power constraint and reformulate the problem only with b_1 and b_2.

c. (Extension to two tone case) Finally, extend the problem to a two tone case. The channel in the n^{th} tone for the u^{th} user is defined as follows.

$$Y_{u,n} = H_{u,n} X_n + N_{u,n} \quad (u = 1, 2 \ \& \ n = 1, 2) \tag{14.88}$$

In both tones, suppose that user 1 always has the better channel than user 2, i.e., $|H_{1,n}| > |H_{2,n}|$ (n=1,2). Thus, user 1 is decoded last in both tones. Similar to (b), the attempt to find the capacity region boundary is equivalent to solving the following optimization problem as we sweep the variable a_1 and a_2.

$$\begin{array}{ll} \max & a_1 b_1 + a_2 b_2 \\ \text{s.t} & b_1 \le \frac{1}{2}\log_2(1 + |H_{1,1}|^2 \mathcal{E}_{1,1}) + \frac{1}{2}\log_2(1 + |H_{1,2}|^2 \mathcal{E}_{1,2}) \\ & b_2 \le \frac{1}{2}\log_2\left(1 + \frac{|H_{2,1}|^2 \mathcal{E}_{2,1}}{1 + |H_{2,1}|^2 \mathcal{E}_{1,1}}\right) + \frac{1}{2}\log_2\left(1 + \frac{|H_{2,2}|^2 \mathcal{E}_{2,2}}{1 + |H_{2,2}|^2 \mathcal{E}_{1,2}}\right) \\ & \mathcal{E}_{1,1} + \mathcal{E}_{1,2} + \mathcal{E}_{2,1} + \mathcal{E}_{2,2} \le 1, \end{array}$$

Reformulate this problem as a geometric programming form. (4 pts)

Hint. Think of this as a virtual 4-user-single-tone case.

14.12 *Rate Sum Maximization and Duality (10 pts).*

Reconsider the vector BC that was introduced in Section 14.2's example, which is repeated here for readers' convenience. The channel parameters for this example are $L_x = 3$,$L_y = 1$,$U = 3$,$N = 1$ and $\mathcal{E}_{\boldsymbol{x}} = 10$.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \boldsymbol{y} = H\boldsymbol{x} + \boldsymbol{n} = \begin{bmatrix} 2 & 1 & .5 \\ 1 & 3 & .2 \\ 2 & .1 & 4 \end{bmatrix} \mathbf{x} + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix} \tag{14.89}$$

This can be rewritten as $y_i = H_i \boldsymbol{x} + n_i \ (i = 1, 2, 3)$,

where $\sigma_{n_i}^2 = 1$ and H_i's are the row vectors such that $H = \begin{bmatrix} H_1 \\ H_2 \\ H_3 \end{bmatrix}$.

a. Write a dual MAC channel for this vector BC. (2 pts)

b. Using the program you wrote in Problem 13.10, find out the maximum rate sum for this dual MAC and the optimal power distribution among users. Also determine this sum rate for the situation of a maximum sum of energies.(4 pts)

c. Using MAC-to-BC transformation in Section 14.3.2, transform the result you found in part b to BC and compare the result with what you found in Problem 14.9. (4 pts)

14.13 *Worst case noise calculation: (10 pts).*

Consider the same Broadcast channel of problem 14.10. In this problem, the structure of worst-case noise corresponding to maximum rate-sum point is investigated through duality between MAC and BC. This problem exploits the results of problem 14.4 and the algorithm devised in problem 13.10 of chapter 13 to compute the worst case noise. To remind you, worst case noise is a zero mean Gaussian noise with covariance matrix S_z that minimizes $I(\mathbf{x}; \mathbf{y})$ for the following point to point channel

$$\boldsymbol{y} = \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} \boldsymbol{x} + \boldsymbol{n}$$

subject to $R_{\boldsymbol{nn}}(i) = I$. Where $R_{\boldsymbol{nn}}(i)$ is the i^{th} diagonal sub-block of the matrix $R_{\boldsymbol{nn}}$. This point to point channel is obtained from original BC by allowing coordination at the receivers' side. Note that worst case noise is defined for any transmit covariance matrix $R_{\boldsymbol{xx}}$, however this problem considers the one corresponding the optimal transmit covariance matrix S_x that maximizes the mutual information term.

a. Define the matrix Q exactly the same as the one in problem 14.10. Show that capacity of the point-to-point channel introduced in problem 14.10 is an upper bound on capacity of the point-to-point channel defined in this problem with the noise covariance matrix given as,

$$R_{\boldsymbol{nn}} = \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} Q \begin{bmatrix} H_1 \\ H_2 \end{bmatrix}^* .$$

b. Prove that capacity of the point to point channel defined here with $R_{\boldsymbol{nn}}$ of part (a) is still an upper bound on the sum-capacity of the original broadcast channel. Hence, sum-capacity of the original BC is a lower bound on this capacity. However, this lower bound is equal the upper bound of part (a) based on the result of problem 14.10. Therefore, capacity of this point-to-point channel is equal to sum-capacity of BC. Note that for given $R_{\boldsymbol{nn}}$, $R_{\boldsymbol{nn}}(i)$, diagonal sub-block matrices, are not necessarily equal to identity matrix, however it can be shown that, if these sub-block matrices be replaced by identity matrix the worst case noise is obtained.

c. (*Bonus*) Prove the claim that by replacing $R_{\boldsymbol{nn}}(i) = H_i Q H_i^*$ with I, $I(\boldsymbol{x}, \boldsymbol{y})$ does not change and the worst case noise is obtained.

d. Based on what is proved in previous parts, use the algorithm of problem 13.10 to find the matrix Q and the worst case noise for example of Section 14.2.