Contents

14 The Gaussian Vector Broadcast Channel 441
  14.1 The Vector Broadcast Channel 442
  14.1.1 Modeling the BC 442
  14.1.2 Forney’s Crypto Precoder 443
  14.2 Worst-Case Noise and BC Rate Sums (WCN) 446
    14.2.1 WCN GDFE diagonalization 446
    14.2.2 Yu’s Maximum Rate Sum for the BC 457
  14.3 Vector Duality for the BC 459
    14.3.1 Channel Equivalences 459
    14.3.2 Duality 463
    14.3.3 Determination of $R_{\Delta X}$ for given rate tuple 468
    14.3.4 The dual-channel’s GDFE 469
  14.4 Vector DMT and the BC 474
    14.4.1 VDMT for the BC 474
    14.4.2 Design Assessment 476
  14.5 Generation of the BC Capacity Rate Region 479
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’ 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 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 input realization, and then consequently to the corresponding best BC implementation. Each user in the BC has its own separate MMse-GDFE that can be used to construct a precoder-coefficient set for that user. The superset of all users’ precoder-coefficient sets defines a triangular precoder for the BC.

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 such worst-case noise 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’s maximum rate sum. A caution in the use of worst-case noise also occurs in the form of a simple $1 \times 2$ 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. A set of dual-GDFEs for the BC then specifies precoder design of the best BC transmitter (and somewhat trivial corresponding receiver). Section 14.4 then concludes with the Vector-DMT approach for the BC.

The determination of the best input autocorrelation for any point $b \in c(b)$ 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. This chapter largely considers zero gap; the methods for non-zero gap and duality remain an open problem at time of writing.
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 $x$ has dimensionality $L_x(N + \nu) \times 1$ and is the sum of $U$ independent components for the Gaussian channel

$$x = \sum_{u=1}^{U} x_u .$$  \hspace{1cm} (14.1)

$$y = \begin{bmatrix} y_U \\ \vdots \\ y_1 \end{bmatrix} = \begin{bmatrix} H_U \\ \vdots \\ H_1 \end{bmatrix} x + \begin{bmatrix} n_U \\ \vdots \\ n_1 \end{bmatrix}$$  \hspace{1cm} (14.2)

$$y = H x + n .$$  \hspace{1cm} (14.3)

The individual components $y_u$ of $y$ 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 \( n \). The input autocorrelation matrix is

\[
R_{xx} = \sum_{u=1}^{U} R_{xx}(u) .
\]  

(14.4)

A single overall transmit-power constraint is expressed as

\[
\text{trace}\{R_{xx}\} \leq E_x .
\]  

(14.5)

Each of the BC outputs can have its own GDFE that may attempt to estimate \( x_u \) and/or any subsets of the other users. When a simple ordering of the BC channels in terms of their gains is possible, then GDFE’s may use feedback sections at the receiver outputs to estimate those other users that can be decoded first. When such simple ordering is possible, as for instance is always the case on the \( 1 \times U \) Gaussian BC, then receiver GDFE feedback-section implementation is possible. These are often called **degraded broadcast channels**, where a certain ordering is inferred in that previous users can all be decoded. More generally, some dimensions of user \( u \) may be decodable by receiver \( u' \neq u \), while other dimensions of user \( u \) may not be decodable. In such non-degraded broadcast channels, a transmitter precoder is used in place of a receiver feedback section.

### 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 \( v_u \) over Lattice \( \Lambda_u \)’s Voronoi region \( V(\Lambda_u) \): The output \( x_u \) of the modulo device is then independent of the input \( v_u \) and of the added signal \(-\sum_{i=u+1}^{U} g_{u,i} \cdot x_i\), and more importantly this output has the same energy as the input \( v_u \), \( E_{v_u} = E_{x_u} \). The result holds for \( g_{u,i} \) equal to any set of coefficients. The second adder at the input to the channel adds the other users’ signals and its user components, and this second adder may be viewed as adding the “side” information for any of the other signals that occur earlier in an ordering of users. User \( u \) views users \( 1, \ldots, u-1 \) 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.

![Forney’s Crypto Precoder](image)

Figure 14.2: Forney’s Crypto Precoder.
Figure 14.3: GDFE Precoder.

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

\[
A = [A_1 \ldots A_2 A_1]
\]  

(14.6)

Each of the sets of precoder coefficients could be viewed as rows of a triangular precoder matrix of a GDFE. Thus, Figure 14.3 uses the GDFE-like description with a “white” input \( \mathbf{u} \) that has components (each on its own dimensions \( \mathbf{v}_u \), \( u = 1, \ldots, U \)). Feedback in this precoder implementation is on each of the successive user inputs to the transmit matrix \( A \). The input autocorrelation is

\[
R_{\mathbf{xx}} = \sum_{u=1}^{U} R_{\mathbf{xx}}(u) = \sum_{u=1}^{U} A_u R_{\mathbf{uu}}(u) A_u^* = AR_{\mathbf{uu}}A^*
\]  

(14.7)

where \( A = \text{diag}\{A_1 \ldots A_2\} \) and \( R_{\mathbf{uu}} = \text{diag}\{R_{\mathbf{uu}}(U) \ldots R_{\mathbf{uu}}(1)\} \). 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 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 \( \Lambda_u \) modulo (and associated matrix multiply \( A_u \)) operation forces a Gaussian output characterized by autocorrelation \( R_{\mathbf{xx}}(u) \). In the scalar case of \( L_x(N + \nu) = 1 \), then the modulo \( \Lambda_u \) could be viewed as a reflection about an infinite-dimensional hypersphere of radius

\[
444
\]
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 with characteristic $R_{xx}^{1/2}(u)$ where any square root is allowed - such transmit matrices are generally denoted by $A_u$ as in Figure 14.3, where $A = [A_U \ldots A_1]$. Figure 14.3 also shows the $U$ receivers that each allow processing of the received signals independently. Each user can be interpreted in terms of its own GDFE, and there is also an overall GDFE.

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 modulo/circular quantizer) 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-$\Lambda_u$ device at receiver $y_u$, 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:

$$A'(b, \pi) = \left\{ b \mid 0 \leq b_u \leq \frac{1}{2} \log \left[ \frac{\sum_{i=1}^{U} H_x R_{xx}(i) H_x^* + R_{nn}(i)}{\sum_{i=1}^{u-1} H_x R_{xx}(i) H_x^* + R_{nn}(i)} \right] \right\}.$$  \hspace{1cm} \text{(14.8)}$$

where the non-causal precoder is used to pre-subtract those users who are later in the order. Each user’s rate $b_u$ is thus upper-bounded by the rate of a MMSFE-GDFE system corresponding to $I(x_u; y_u/x_u+1\ldots x_U)$. This GDFE is a function of $u$ (unlike the MAC where the GDFE is for $I(x_u; y/x_u+1\ldots x_U)$), that is $y$ is not a function of $u$). 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 $R_{\text{noise}}(u)$ 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

$$A(b) = \bigcup_{\pi} A'(b, \pi) .$$ \hspace{1cm} \text{(14.9)}$$

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

$$c(b) = \bigcup_{\pi} A(b) \quad \{R_{xx}(u)\} \quad \sum_{u=1}^{U} \text{trace}\{R_{xx}(u)\} \leq \mathcal{E}_x .$$ \hspace{1cm} \text{(14.10)}$$

\footnote{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 $R_{nn}$ is optimized over the off-diagonal terms while the diagonal noise-power terms are held constant to minimize a channel’s mutual information for a given $R_{xx}$. For the BC case, the diagonal terms are actually $L_y N \times L_y N$ block element matrices $R_{nn}(u)$ 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 $R_{nn}$. Those off-diagonal blocks are considered variable in WCN determination.

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 $R_{nn}$. Thus, to use that software, the individual user channels have to be pre-whitened to be

$$H_u = R_{nn}^{1/2}(u) \cdot H_u$$

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 with 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 when $H$ is full rank:

- Given $R_{xx}$ and $H$, where $\rho(H) = \rho(R_{wcn}) = UNL_y$,
  1. compute $R_{wcn}$ via $[R_{wcn,bwcn}] = wcnoise(Rxx, H, Ly)$
  2. $D = R_{wcn}^{-1} - [HR_{xx}H^* + R_{wcn}]^{-1}$, block diagonal $D \geq 0$.
  3. $[0R]Q^* = R_{wcn}^{-1}H$ via QR factorization
  4. $Q = [q_1 Q_1]$, where if $H$ is square, $Q_1 = Q$ and $q_1 = \emptyset$.
  5. $UU^* = Q_1^* R_{xx} Q_1$, 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. $S_0^{-1} = D_A^{-1} DD_A^{-1} = SNR$ (or simply $S_0^{-1}(i) = \frac{D(i)}{D_A'(i)}$).

---

$^2$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 $HR_{xx}H^* = HAA^*H^*$ is a check on this value of $A$, and indeed for this zero-null-space construction of $R_{xx}$, then $R_{xx} = AA^*$, although when $H$ is singular, only $HR_{xx}H^* = HAA^*H^*$ may hold.
10. $W^{unb} = (SNR - I)^{-1}G^* A^* H^* R_{wc}^{-1} = (SNR - I)^{-1} D_A$ (diagonal feed-forward filter\(^4\)).

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

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

**EXAMPLE 14.2.1 (Worst-Case Noise for square Channel with GDFE)**

\[
\begin{bmatrix}
0 & 1.0000 & 0.5000 \\
0.3000 & 0.6000 & 1.0000
\end{bmatrix}
\]

\[
\begin{bmatrix}
1.0000 & 0.8000 & 0.6400 \\
0.8000 & 1.0000 & 0.8000 \\
0.6400 & 0.8000 & 1.0000
\end{bmatrix}
\]

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

\[
Rwcn = \begin{bmatrix} 1.0000 & 0.3871 & 0.4898 \end{bmatrix}
\]

\(^4\)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 $U$. 

447
\[
\begin{bmatrix}
0.3871 & 1.0000 & 0.6541 \\
0.4898 & 0.6541 & 1.0000 \\
\end{bmatrix}
\]

\[
b_{\text{max}} = 1.4336
\]

\[
D = \text{inv}(R_{\text{wcn}}) - \text{inv}(H \cdot R_{\text{xx}} \cdot H' + R_{\text{wcn}})
\]

\[
\begin{bmatrix}
0.4847 & 0.0000 & 0.0000 \\
0.0000 & 0.1985 & 0.0000 \\
0.0000 & 0.0000 & 0.4675 \\
\end{bmatrix}
\]

\[
H_{\text{tilde}} = \text{inv}(R_{\text{wcn}}) \cdot H
\]

\[
\begin{bmatrix}
1.1643 & 0.1798 & -0.2285 \\
-0.4787 & 1.0409 & -0.2427 \\
0.0428 & -0.1690 & 1.2707 \\
\end{bmatrix}
\]

\[
J_3 = \text{hankel}([0 \ 0 \ 1]);
\]

\[
[Q, R] = \text{qr}(J_3 \cdot H_{\text{tilde}}' \cdot J_3);
\]

\[
Q = (J_3 \cdot Q \cdot J_3) =
\begin{bmatrix}
0.9059 & 0.4221 & -0.0334 \\
0.4226 & -0.8967 & 0.1317 \\
0.0257 & -0.1334 & -0.9907 \\
\end{bmatrix}
\]

\[
R = (J_3 \cdot R \cdot J_3)' =
\begin{bmatrix}
1.1250 & 0.3607 & 0.2112 \\
0 & -1.1030 & 0.3936 \\
0 & 0 & -1.2826 \\
\end{bmatrix}
\]

\[
R_{\text{xxrot}} = Q' \cdot R_{\text{xx}} \cdot Q =
\begin{bmatrix}
1.6598 & -0.6412 & -0.8230 \\
-0.6412 & 0.5138 & 0.5003 \\
-0.8230 & 0.5003 & 0.8264 \\
\end{bmatrix}
\]

\[
U = (J_3 \cdot \text{chol}(J_3 \cdot R_{\text{xxrot}} \cdot J_3) \cdot J_3)' =
\begin{bmatrix}
0.8621 & -0.3113 & -0.9053 \\
0 & 0.4593 & 0.5503 \\
0 & 0 & 0.9091 \\
\end{bmatrix}
\]

\[
D_A = \text{diag}(\text{diag}(R \cdot U)) =
\begin{bmatrix}
0.9699 & 0 & 0 \\
0 & -0.5067 & 0 \\
0 & 0 & -1.1660 \\
\end{bmatrix}
\]

\[
G = \text{inv}(D_A) \cdot R \cdot U =
\begin{bmatrix}
1.0000 & -0.1902 & -0.6475 \\
\end{bmatrix}
\]
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 $I_{\text{wcn}}$. 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 $I_{\text{wcn}}$ 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, $I(x_u; y_u|x_1 \ldots x_{u-1})$, cannot be improved and is achieved by any (block) “scalar” multiplication and a subsequent maximum-likelihood (or mod-$\Lambda_u$) 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 $\frac{|R_{xx}|}{|\tilde{R}_{\text{noise}}(u)|}$) cannot be improved without cooperation 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 $R_{xx}$ would have been obscured). QED.

The determination of the diagonalized GDFE receiver can be somewhat simplified when $\rho(H) < NL_y U$; 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

$$R_{wcn}^{-1}H A G^{-1} S_0^{-1} G^{-*} A^* H^* R_{wcn}^{-1} = D$$

(14.15)

where $D$ is the same diagonal matrix that characterizes the original worst-case noise equation in (14.13). 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,

$$R_{wcn}^{-1} = \begin{bmatrix} R & 0 \\ 0 & 0 & I \end{bmatrix} Q^*.$$

(14.16)

The choice of $A$ that solves (14.15) is

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

(14.17)

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

1. compute $R_{wcn}$ via $[R_{wcn}, b_{wcn}] = \text{wnoise}(R_{xx}, H, Ly)$

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

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. $UU^* = Q^* R_{xx} Q$, 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)

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

450
Figure 14.5: Simple BC example.

8. \[ \begin{bmatrix} S_0^{-1} & 0 \\ 0 & 0 \end{bmatrix} = D_A^{-1} D_A^{-1} = \begin{bmatrix} SNR & 0 \\ 0 & 0 \end{bmatrix} \] (or simply \( S_0^{-1}(i) = \frac{d(i)}{\mathcal{E}_A(i)} \), \( i = 1, \ldots, \rho(H) \)).

9. \( W_{\text{unb}} = \begin{bmatrix} SNR - I & 0 \\ 0 & 0 \end{bmatrix} D_A \) (feedforward filter).

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

The simple 1 \( \times \) 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 \( \mathcal{E}_x = 1 \). The capacity region re-appears 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:

<table>
<thead>
<tr>
<th>( \mathcal{E}_1 )</th>
<th>( \mathcal{E}_2 )</th>
<th>( b_1 )</th>
<th>( b_2 )</th>
<th>( b_1 + b_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>0</td>
<td>6.32</td>
<td>0</td>
<td>6.32</td>
</tr>
<tr>
<td>.75</td>
<td>.25</td>
<td>6.12</td>
<td>.20</td>
<td>6.32</td>
</tr>
<tr>
<td>.50</td>
<td>.50</td>
<td>5.82</td>
<td>.50</td>
<td>6.32</td>
</tr>
<tr>
<td>.25</td>
<td>.75</td>
<td>5.32</td>
<td>1.0</td>
<td>6.32</td>
</tr>
<tr>
<td>.10</td>
<td>.9</td>
<td>4.66</td>
<td>1.66</td>
<td>6.32</td>
</tr>
<tr>
<td>.05</td>
<td>.95</td>
<td>4.16</td>
<td>2.20</td>
<td>6.26</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>5.64</td>
<td>5.64</td>
</tr>
</tbody>
</table>

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).

A subtle problem can arise when individual data rates, and not just rate sum, are of interest. The following examples illustrate this problem.

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

\[ R_{\text{wcn}} = \begin{bmatrix} 1 \\ 5/5 \end{bmatrix} \]  

\[(14.18)\]
This $R_{\text{wnc}}$ 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 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 $y$, even though the GDFE feedforward section is diagonal and corresponding to worst-case noise, only measures the maximum sum rate that can be achieved by all users to all the receivers, including in particular all users to receiver 1.

Thus, while worst-case noise and the GDFE are powerful concepts that simplify greatly multi-user channels, their use does not 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.

```matlab
>> 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*Q*J2)';
>> R=R*J2 =

    -113.1369   -0.0000
     -0.0002       0

>> Q=Q*J2 =

452
Figure 14.6: Equivalent Broadcast channel with summer viewed in channel.

\[
\bar{H} = \begin{bmatrix}
50 & 50 \\
80 & 80
\end{bmatrix}
\]
\[-0.7071 \quad 0.7071 \\
-0.7071 \quad -0.7071\]

\[R^*Q'-H_{\text{tilde}} = 1.0e-013 *\]

\[
\begin{bmatrix}
0 & -0.1421 \\
0 & 0
\end{bmatrix}
\]

\[U = (J_2 \cdot \text{chol}(J_2 \cdot R_{xx\text{rot}} \cdot J_2) \cdot J_2)' =\]

\[
\begin{bmatrix}
0.7071 & 0 \\
0 & 0.7071
\end{bmatrix}
\]

\[r = R(1,1) = -113.1369\]

\[R_{\text{new}} = [r \ 0 \\
0 \ 1] =\]

\[
\begin{bmatrix}
-113.1369 & 0 \\
0 & 1.0000
\end{bmatrix}
\]

\[DA = \text{diag}(\text{diag}(R_{\text{new}} \cdot U)) =\]

\[
\begin{bmatrix}
-79.9999 & 0 \\
0 & 0.7071
\end{bmatrix}
\]

\[G = \text{inv}(DA) \cdot R_{\text{new}} \cdot U =\]

\[
\begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}
\]

\[A = Q \cdot \text{inv}(R_{\text{new}}) \cdot DA \cdot G =\]

\[
\begin{bmatrix}
-0.5000 & 0.5000 \\
-0.5000 & -0.5000
\end{bmatrix}
\]

\[S_{0\text{inv}} = \text{inv}(DA) \cdot D \cdot \text{inv}(DA);\]

\[S_{0} = \begin{bmatrix} 1/S_{0\text{inv}}(1,1) & 0 \\
0 & 0 \end{bmatrix} =\]

\[
1.0e+003 *
\begin{bmatrix}
6.4010 & 0 \\
0 & 0
\end{bmatrix}
\]

\[W_{\text{unb}} = \text{diag}([1/(S_{0}(1,1)-1) \ 0]) \cdot DA =\]

\[
\begin{bmatrix}
-0.0125 & 0 \\
0 & 0
\end{bmatrix}
\]

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 unassisted GDFE in this case is still attempting to estimate that maximum rate. The solution is instead to find a GDFE for each user.
EXAMPLE 14.2.3 (simple BC via dual GDFEs) For the same BC channel, this example computes the GDFE’s for each receiver for the 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, and energy per user and bit rates were there obtained.

```
>> H = 80 80 (user 1 is on top/left for receiver 1, user 2 known in GDFE)
>> Rxx= 0.5000 0
    0 0.5000
>> Rinv=H'*H+inv(Rxx) = 6402 6400
    6400 6402
>> Gbar=chol(Rinv) = 80.0125 79.9875
    0 1.9998
>> G=inv(diag(diag(Gbar)))*Gbar = 1.0000 0.9997
    0 1.0000
>> S0=diag(diag(Gbar))=diag(diag(Gbar)) =

    1.0e+003 * 6.4020 0
    0 0.0040
>> b=.5*log2(diag(Rxx*S0)) = 5.8222
    0.4999
>> SNR=Rxx*S0 = 1.0e+003 * 3.2010 0
    0 0.0020
>> Wub=SNR*inv(SNR-eye(2))*inv(S0)*inv(G')*H'=
```

Figure 14.7: Rate region for broadcast channel

455
Receiver 1 is in the preferred position so that user 2 in this degraded broadcast channel can be decoded first at a GDFE for receiver 1. Equivalently, and not dependent on the channel being degraded broadcast, a precoder could be used to remove user 2's effect on user 1 at receiver 1. The feedback coefficient for user 2 into user 1 (whether implemented at receiver 1 or as a precoder) is 1. Receiver 2 must treat user 1 as noise. A second GDFE is designed for this situation in the latter half of the matlab example above. In this case only the bottom row of G and W are important and do indeed correspond.
to treating user 1 as noise. The top row of receiver 2’s GDFE is not implemented.

The transmit energies used in these GDFEs are the broadcast channel energies. The bit rates correspond to the bit rates in the dual MAC (which has a dual set of energies as in Chapter 12). More generally, the dual-channel MAC can be used to find admissible bit rates and corresponding MAC and thus through duality BC energies and equal bit rates per user. From Chapter 12, these energies were \(1/2\) and \(1/2\) and the two users’ data rates were 5.8 and 0.5. Receiver 1’s GDFE attains these rates, while the rate of User 2 in receiver 2’s GDFE is correctly 0.5. If Receiver 2 had desired to compute user 1’s data rate, it would not be possible since 5.8 exceeds the 5.1 that would be possible. If the transmitter somehow sent less information for user 1 of 5.1, then user 1 could be decoded at receiver 2 and the GDFE shown would extract that lower data rate.

The precoder for user 2 into user 1 is determined by the unbiased feedback coefficient of Receiver 1’s GDFE. More generally, the feedback coefficients for user \(u + j\)’s effect into user \(u\) is determined by the feedback coefficients for users \(u + 1 \ldots U\) for Receiver \(u\)’s GDFE. Thus one set of feedback coefficients from each of Receivers 1, ..., U-1 is used to construct the overall GDFE precoder, or an overall GDFE. This overall GDFE will correspond to a worst-case noise for the broadcast channel, but is computed in an incremental channel via duality without need for the worst-case noise autocorrelation.

### 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 \(R_{xx}\) for which the \(R_{wcn}\) equation is satisfied. Problem ?? further develops software for the simple convergent process of

1. Initialize \(R_{xx} = \frac{\mathcal{E}^s}{\mathcal{N}} \cdot I\).
2. Compute the \(R_{wcn}\) for the given \(R_{xx}\).
3. Compute the water-filling \(R_{xx}\) 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 \(R_{xx}\) 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), as illustrated in Figure 14.8.
Figure 14.8: Convergence of BC rate-sum algorithm.
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 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 user’s rates equal to those of the dual MAC 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 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

\[ y = Hx + n \]

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 “fat” matrix \( H \) is augmented by \( l_x - l_y \) new zeroed rows of \( l_x \) zeros each, at the bottom of the matrix:

\[ H \rightarrow \begin{bmatrix} \hat{H} \\ 0 \end{bmatrix} \]

(14.20)

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

\[ y \rightarrow \begin{bmatrix} y \\ \text{don’t care} \end{bmatrix} \quad ; \quad n \rightarrow \begin{bmatrix} n \\ \text{don’t care} \end{bmatrix} \]

(14.21)

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

\[ R_{nn} \rightarrow \begin{bmatrix} R_{nn} & 0 \\ 0 & I \end{bmatrix} \]

(14.22)

The noise equivalent channel is then

\[ \hat{H} \rightarrow \begin{bmatrix} R_{nn}^{-1/2}H \\ 0 \end{bmatrix} = \begin{bmatrix} R_{nn} & 0 \\ 0 & I \end{bmatrix}^{-1/2} \cdot \begin{bmatrix} H \\ 0 \end{bmatrix} \]

(14.23)

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

When \( l_y > l_x \), some zeroed artificial input columns are created to make the “tall” matrix \( H \) square \( l_y \times l_y \). The new channel matrix becomes

\[ H \rightarrow [H \ 0] \]

(14.24)
with \( l_y - l_x \) extra columns of \( l_y \) zeros each. The output \( y \) and noise \( n \) remain \( l_y \times 1 \) vectors. The input \( x \) has an extra \( l_y - l_x \) zero elements at the bottom (these could be anything because of the zeros in the channel, but a choice of zero keeps energy to a minimum on the extra dummy dimensions). The input autocorrelation matrix becomes

\[
R_{xx} \rightarrow \begin{bmatrix} R_{xx} & 0 \\ 0 & 0 \end{bmatrix}.
\] (14.25)

The noise-equivalent channel is

\[
\bar{H} = R_{nn}^{-1/2} \cdot H.
\] (14.26)

The mutual information remains \( I(x; y) \).

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

\[
I(x; y) = \frac{1}{2} \log_2 | I + \bar{H} \cdot R_{xx} \cdot \bar{H}^* | .
\] (14.27)

**Theorem 14.3.1 (Dual Channels)** The square channel with white noise has a dual that is shown in Figure 14.9, along with the GDFE for each. Both these channels have the same mutual information

\[
I(x; y) = I(\bar{x}; \bar{y})
\] (14.28)

and the same input energy

\[
\text{trace}(R_{xx}) = \text{trace}(\bar{R}_{xx}).
\] (14.29)

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

\[
\bar{H} = F \cdot \Lambda \cdot M^*.
\] (14.30)

\[\text{Calculations for normalized quantities to the number of dimensions should continue to use } l_x \text{ and not } l_y.\]
Then
\begin{align*}
y &= F \cdot \Lambda M^* \cdot x + n \quad (14.31) \\
y' &\triangleq F^* y = \Lambda \cdot (M^* \cdot x) + n' \\
&\triangleq \Lambda \cdot F^* \cdot (F \cdot M^* \cdot x) + n' \\
&\triangleq \Lambda \cdot F^* \cdot \bar{x} + n' \quad (14.33) \\
y' &\triangleq M y' = M \cdot \Lambda \cdot F^* \cdot \bar{x} + \bar{n} \quad (14.35) \\
\bar{y} &= \bar{H}^* \cdot \bar{x} + \bar{n} \quad (14.36)
\end{align*}

The noise $\bar{n}$ also has an identity 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 (14.31) - (14.36), all transformations were orthogonal and 1-to-1 so the mutual information is preserved. Equation (14.36) corresponds to the dual channel in Figure 14.9. Thus,
\begin{equation}
I(x; y) = I(\bar{x}; \bar{y}) \quad (14.37)
\end{equation}
\begin{equation}
\frac{1}{2} \log_2 | I + \bar{H} R_{xx} \bar{H}^* | = \frac{1}{2} \log_2 | I + \bar{H}^* R_{xx} \bar{H} | \quad .
\end{equation}

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
\begin{equation}
\bar{x} = F \cdot M^* \cdot x \quad .
\end{equation}

The input autocorrelation matrices are related by
\begin{equation}
R_{xx} = F \cdot M^* \cdot R_{xx} \cdot M \cdot F^* \quad .
\end{equation}

Because both of the transformations in (14.39), $F$ and $M^*$ are orthogonal, they are energy preserving and so the trace remains the same.

For a given transmit covariance $R_{xx} = A_x A_x^*$, the GDFE for the original channel is computed by Cholesky factorization of
\begin{equation}
R_b^{-1} = A_x^* \bar{H} \bar{H}^* A_x + I = \bar{G}^* S_0 G \quad ,
\end{equation}
while for the dual channel this factorization is for
\begin{align*}
\bar{R}_b^{-1} &= \bar{A}_x^* \bar{\bar{H}} \bar{\bar{H}}^* \bar{A}_x + I \\
&= FM^* A_x^* \bar{H} \bar{H}^* A_x MF^* + I \\
&= FM^* G^* S_0 G MF^* \quad .
\end{align*}

QED.

When $l_x > l_y$, it is possible that the input $x$ and consequently its autocorrelation matrix $R_{xx}$ 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 is
\begin{equation}
\mathcal{E}_x = \mathcal{E}_x(\text{pass}) + \mathcal{E}_x(\text{null}) \quad (14.45)
\end{equation}
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 $F$ (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,
\begin{equation}
F^* \bar{x} = M^* x \quad (14.46)
\end{equation}
still holds. The quality on the right \( M^*x \) only contains components in the pass space of \( \tilde{H} \). Computing the squared norm of the vector written in (14.46)

\[
\tilde{x}^*FF^*\tilde{x} = x^*MM^*x
\]

\[
\|\tilde{x}\|^2 = \|x\|^2_{\text{pass}}.
\]

Thus, duality can be executed with only the \( \rho(\tilde{H}) \) 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 \( \rho(\tilde{H}) \times l_x \) matrix of columns formed an used in (14.46). The nonsingular columns of \( M \) remain the same and are still \( l_x \) dimensional in both singular value decompositions (of square \( \tilde{H} \) and non-square \( \tilde{H} \)). However, (14.46) 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 \( \tilde{H} \) being square. The dual channel and the original channel have the same mutual information and the same input energies, as long as no energy is wasted in the null space.

The dual channels could be viewed with \( \tilde{H} = \begin{bmatrix} \tilde{H}_U \; : \; \tilde{H}_1 \end{bmatrix} \) as a BC and \( H^* = [\tilde{H}_U^* \; \ldots \; \tilde{H}_1^*] \) as a MAC. Each user of the BC has a channel element \( \tilde{H}_u \) associated with it, and consequently the dual MAC user has a corresponding dual channel \( \tilde{H}_u^* \). 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, when the MAC input \( x^M = FM^*x^B \), relates that the sum of the BC users’ energies must equal the sum of the energies on the input to the MAC channel,

\[
\sum_{u=1}^U \mathcal{E}_u^N = \sum_{u=1}^U \mathcal{E}_u^M.
\]

However, the individual user channels each 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 of interest include “other-user” noise (as in the MAC’s successive decoding or the BC’s precoding) in calculations involving \( \tilde{H}_u \) and those other user noises are not yet included in the simple duality that occurs from the overall channel’s conjugate transpose. In fact, there is no guarantee that an input \( \tilde{R}_{xx} \) 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. Noise whitening \( R_n^{1/2} \cdot \tilde{H} \) on a channel corresponds to input deflection by \( \tilde{R}_{nn}^{1/2} \) on its dual \( \tilde{H}^* \cdot R_{nn}^{1/2} \).

The following theorem is quite general, but will be used in the context of the deflecting matrix being the square root of a \( \tilde{R}_{\text{noise}}(u) \) matrix for a given order in duality between a MAC and a BC.
Theorem 14.3.2 (Input Deflection with Correlated Noise in Duality) The two dual individual-user 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 multiplication does not change the mutual information. Or directly mathematically noting simply that there is a multiplication by an identity does not change the expression,

\[
\bar{H} \cdot R_{nn}^{-1/2} \left( R_{nn}^{1/2} \cdot R_{xx} \cdot R_{nn}^{1/2} \right) \cdot R_{nn}^{-1/2} \bar{H}^* = \bar{H} \cdot R_{xx} \cdot \bar{H}^* .
\]

(14.50)

The mutual information remains the same\(^6\), namely

\[
I(\tilde{x}; y) = I(x; y) ,
\]

(14.51)

when

\[
\tilde{x} = R_{nn}^{1/2} \cdot x .
\]

(14.52)

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

\[
\tilde{H} = \bar{H} \cdot R_{nn}^{1/2} .
\]

(14.53)

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

\[
\tilde{H} = \tilde{F} \cdot \tilde{\Lambda} \cdot \tilde{M}^* ,
\]

(14.54)

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

\[
I(\tilde{\tilde{x}}; \tilde{y}) = I(\tilde{x}; y) = I(x; y) ,
\]

(14.55)

where the input transformation is

\[
\tilde{\tilde{x}} = \tilde{F} \cdot \tilde{M}^* \cdot \tilde{x} = \tilde{F} \cdot \tilde{M}^* \cdot R_{nn}^{1/2} \cdot x .
\]

(14.56)

The autocorrelation matrix relationships are

\[
R_{\tilde{x}\tilde{x}} = \tilde{F} \cdot \tilde{M}^* \cdot R_{xx} \cdot \tilde{M} \cdot \tilde{F}^* = \tilde{F} \cdot \tilde{M}^* \cdot R_{nn}^{1/2} \cdot R_{xx} \cdot R_{nn}^{1/2} \cdot \tilde{M} \cdot \tilde{F}^* .
\]

(14.57)

This second use of dual channels absorbs the user-dependent \(\tilde{R}_{\text{noise}}\) terms that occur in the BC and MAC. QED.

The key point is that input deflection does not change the bit rate \(I(x; y)\). This concept of input deflection and the dual channel will be applied not to the overall channel \(\bar{H}\) where 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 14.12 illustrates a general BC and its MAC dual. Theorem 14.3.2’s translations can be applied 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), as long as the input transformation and deflection in (14.57) are maintained. However, duality can also be applied at an interior level, in particular with input deflection, to each of the user

\(^6\)The input energy, however, may not be maintained.

463
\[
\tilde{x} = \tilde{F} \cdot \tilde{M}^* \cdot \tilde{x} = \tilde{F} \cdot \tilde{M}^* \cdot R_{nn}^{1/2} \cdot x
\]

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

\[
\begin{align*}
\text{Broadcast} \\
R_{xx}^B(1) & \quad x_1^B \\
R_{xx}^B(2) & \quad x_2^B \\
\vdots & \quad \vdots \\
R_{xx}^B(U) & \quad x_U^B
\end{align*}
\]

\[
\tilde{H} = H^* \cdot R_{nn}^{1/2} \left( u \right)
\]

\[
\tilde{R}_{\text{noise,B}} = I + \sum_{u=1}^{U} \tilde{H}_u \cdot R_{xx}^B(u) \cdot \tilde{H}_u^*
\]

Order reversed

\[
\begin{align*}
\begin{split}
R_{xx}^M(1) \quad x_1^M & \quad \tilde{H}_1^* \\
R_{xx}^M(2) \quad x_2^M & \quad \tilde{H}_2^* \\
\vdots & \quad \vdots \\
R_{xx}^M(U) & \quad x_U^M
\end{split}
\end{align*}
\]

\[
\tilde{R}_{\text{noise,M}} = I + \sum_{u=1}^{U} \tilde{H}_u^* \cdot R_{xx}^M(u) \cdot \tilde{H}_u
\]

\[
\mathbf{E}_x = \sum_u \mathbf{E}_u = \sum_u \text{trace} \left\{ R_{xx}^M(u) \right\}
\]

Figure 14.12: Dual Vector BC and MAC channels.
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 to be 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 consistently reversed from the MAC 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:

$$\hat{R}_{\text{noise},M}(u) = I + \sum_{i=u+1}^{U} \hat{H}_i^* \cdot R_{xx}^M(i) \cdot \hat{H}_i$$  \hspace{1cm} (14.58)

$$\tilde{R}_{\text{noise},B}(u) = I + \tilde{H}_u \cdot \left( \sum_{i=1}^{u-1} R_{xx}^B(i) \right) \cdot \tilde{H}_u^*$$  \hspace{1cm} (14.59)

The individual user and input autocorrelation matrices $R_{xx}^B(u)$ and $R_{xx}^M(u)$ can be chosen so that the data rates on each of the two dual channels are equal, and those data rates are

$$b_u^{MAC} = \frac{1}{2} \log_2 \left| \frac{\hat{H}_u^* \cdot R_{xx}^M(u) \cdot \hat{H}_u + \hat{R}_{\text{noise},M}(u)}{R_{\text{noise},M}(u)} \right|$$  \hspace{1cm} (14.60)

$$b_u^{BC} = \frac{1}{2} \log_2 \left| \frac{\hat{H}_u^* \cdot R_{xx}^B(u) \cdot \hat{H}_u + \tilde{R}_{\text{noise},B}(u)}{R_{\text{noise},B}(u)} \right|$$  \hspace{1cm} (14.61)

These two individual user channels corresponding to the data rates in (14.60) and (14.61) look almost like duals already, 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' noises, as in Equations (14.58) and (14.59). Such input deflection does not change the mutual information by Theorem 14.3.2, so the rates of the two channels with noise-input deflection are maintained equal, and thus (equating the arguments of the logarithms in (14.60) and (14.61))

$$\frac{\left| \hat{H}_u^* \cdot R_{\text{noise},B}^{-1/2}(u) \cdot R_{xx}^M(u) \cdot R_{\text{noise},B}^{-1/2}(u) \cdot \hat{H}_u + \hat{R}_{\text{noise},M}(u) \right|}{\left| R_{\text{noise},M}(u) \right|} = \frac{\left| \hat{H}_u^* \cdot R_{\text{noise},M}^{-1/2}(u) \cdot R_{xx}^M(u) \cdot R_{\text{noise},M}^{-1/2}(u) \cdot H_u^* + \tilde{R}_{\text{noise},B}(u) \right|}{\left| R_{\text{noise},B}(u) \right|}$$ \hspace{1cm} (14.62)

where

$$\tilde{x}_u^M = R_{\text{noise},M}(u) \cdot x_u^M$$  \hspace{1cm} (14.64)

$$\tilde{x}_u^B = R_{\text{noise},B}(u) \cdot x_u^B$$  \hspace{1cm} (14.65)

By dividing the denominator noise into both terms, and defining

$$\tilde{H}_u \triangleq R_{\text{noise},B}^{-1/2}(u) \cdot \hat{H}_u \cdot R_{\text{noise},M}^{-1/2}(u)$$ \hspace{1cm} (14.66)

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

$$| \hat{H}_u^* \cdot R_{xx}^M(u) \cdot \hat{H}_u + I | = | \hat{H}_u^* \cdot R_{xx}^B(u) \cdot \tilde{H}_u^* + I |$$ \hspace{1cm} (14.67)

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

$$\tilde{H}_u = F_u \cdot \Lambda_u \tilde{M}_u^*$$ \hspace{1cm} (14.68)
and a relationship of the two input autocorrelation matrices as

$$ R_{xx}^M(u) = \tilde{F}_u \cdot \tilde{M}_u^* \cdot R_{xx}^B(u) \cdot \tilde{M}_u \cdot \tilde{F}_u^* $$

(14.69)

Furthermore, from the input deflections

$$ R_{xx}^M(u) = \tilde{F}_{\text{noise,B}}(u) \cdot R_{xx}^M(u) \cdot \tilde{F}_{\text{noise,B}}(u) $$

(14.70)

and equivalently

$$ R_{xx}^B(u) = \tilde{F}_{\text{noise,M}}(u) \cdot R_{xx}^B(u) \cdot \tilde{F}_{\text{noise,M}}(u) $$

(14.71)

Combining the equations (14.67), (14.69), and (14.70), the desired relation between MAC and BC covariances is established as (BC to MAC)

$$ R_{xx}^M(u) = \tilde{R}_{\text{noise,B}}^{-\frac{1}{2}}(u) \cdot \tilde{F}_u \cdot \tilde{M}_u^* \cdot \tilde{R}_{\text{noise,M}}^{1/2}(u) \cdot R_{xx}^B(u) \cdot \tilde{R}_{\text{noise,B}}^{1/2}(u) \cdot \tilde{F}_u \cdot \tilde{M}_u \cdot \tilde{R}_{\text{noise,B}}^{-\frac{1}{2}}(u) $$

(14.72)

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

$$ R_{xx}^B(u) = \tilde{R}_{\text{noise,M}}^{-\frac{1}{2}}(u) \cdot \tilde{F}_u \cdot \tilde{M}_u^* \cdot \tilde{R}_{\text{noise,B}}^{1/2}(u) \cdot R_{xx}^M(u) \cdot \tilde{R}_{\text{noise,B}}^{1/2}(u) \cdot \tilde{F}_u \cdot \tilde{M}_u \cdot \tilde{R}_{\text{noise,B}}^{-\frac{1}{2}}(u) $$

(14.73)

The relationships are indexed by $u$. Recursive use of Equation 14.72 must start with user $U$ and successively pass to $U - 1$, ..., 1 because each successive $\tilde{R}_{\text{noise,M}}(u)$ in (14.58) depends on higher indices of $R_{xx}^M(i > u)$. Equation (14.73) must start with user 1 and pass successively to 2, ..., $U$ because each successive $\tilde{R}_{\text{noise,B}}(u)$ in (14.59) depends on lower indices of $R_{xx}^B(i < u)$.

The dual relationships correspond to a recursion that is illustrated in the flow chart of Figure 14.13. In this case $\tilde{R}_{\text{noise,B}}(u)$ 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 (Zou and Mohseni)

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

1. $R_{xx}(u)$ (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. $\tilde{H}^*$, 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. $\pi$ a $1 \times U$ order vector representing the order for the MAC.

There is only one output, which is a tensor of the BC autocorrelation matrices with again the user index last. These may be added together to produce the actual transmit autocorrelation.

% This function converts the covariance 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 dirty paper 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 $k$th 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 $t$ 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

466
Figure 14.13: duality flow chart.

Given:
\[
\begin{align*}
R_{xx}(u) &\text{ for } u = 1, \ldots, U; \quad R_{xx}^U = 0 \\
\tilde{R}_{\text{noise}, \text{M}}(u) = \tilde{R}_{\text{noise}, \text{M}}(1) = 1 \\
\text{BC: } H_u \cdot R_{xx}(u) \\
\text{MAC: } \tilde{H}_{u} = H_u \cdot R_{xx}^{U/(u)}(u)
\end{align*}
\]

For MAC → BC:
\[
\begin{align*}
\tilde{H}_u &= R^{1/2}_{\text{noise}, \text{M}}(u) \cdot \tilde{H}_u \cdot R^{1/2}_{\text{noise}, \text{M}}(u) \\
\tilde{F}_u \cdot \tilde{\Lambda}_u \cdot \tilde{M}_u &= \text{svd} \left( \tilde{H}_u \right) \\
R_{xx}^B(u) &= R_{\text{noise}, \text{B}}^{1/2}(u) \cdot \tilde{M}_u \cdot \tilde{F}_u^* \cdot R_{\text{noise}, \text{M}}^{1/2}(u) \cdot R_{xx}(u) \cdot R_{\text{noise}, \text{M}}^{1/2}(u) \cdot \tilde{F}_u \cdot \tilde{M}_u \cdot R^{1/2}_{\text{noise}, \text{B}}(u) \\
\tilde{R}_{\text{noise}, \text{B}}(u-1) &= \tilde{R}_{\text{noise}, \text{B}}(u) + \tilde{H}_u \cdot R_{xx}^B(u) \cdot \tilde{H}_u^*
\end{align*}
\]

For BC → MAC:
\[
\begin{align*}
R_{xx}^B(u) &= R_{xx}^B(u) \\
\tilde{R}_{\text{noise}, \text{B}}(u) &= \tilde{R}_{\text{noise}, \text{B}}(u) \\
\text{BC: } H_u \cdot R_{xx}(u) \\
\text{MAC: } \tilde{H}_{u} = H_u \cdot R_{xx}^{U/(u)}(u)
\end{align*}
\]

For u = 1, ..., U - 1:
\[
\begin{align*}
R_{xx}^B(u) &= R_{xx}^B(u) + R_{xx}^B(u) \\
\tilde{R}_{\text{noise}, \text{B}}(u) &= \tilde{R}_{\text{noise}, \text{B}}(u) + \tilde{H}_u \cdot R_{xx}^B(u) \cdot \tilde{H}_u^*
\end{align*}
\]

For u = U, ..., 1:
\[
\begin{align*}
\tilde{H}_u &= R^{-1/2}_{\text{noise}, \text{B}}(u) \cdot \tilde{H}_u \cdot R^{-1/2}_{\text{noise}, \text{M}}(u) \\
\tilde{F}_u \cdot \tilde{\Lambda}_u \cdot \tilde{M}_u &= \text{svd} \left( \tilde{H}_u \right) \\
R_{xx}^M(u) &= R^{-1/2}_{\text{noise}, \text{B}}(u) \cdot \tilde{F}_u \cdot \tilde{M}_u \cdot R^{-1/2}_{\text{noise}, \text{M}}(u) \cdot R_{xx}(u) \cdot R_{\text{noise}, \text{M}}^{1/2}(u) \cdot \tilde{M}_u \cdot \tilde{F}_u^* \cdot R^{-1/2}_{\text{noise}, \text{B}}(u) \\
\tilde{R}_{\text{noise}, \text{M}}(u-1) &= \tilde{R}_{\text{noise}, \text{M}}(u) + \tilde{H}_u \cdot R_{xx}^M(u) \cdot \tilde{H}_u^* \text{ skip } u = 1
\end{align*}
\]

Figure 14.13: duality flow chart.
% 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);

% 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 Rtilde noise 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, 'econ');
    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 $R_{xx}$ for given rate tuple

The dual MAC for a BC allows ready determination of the feasibility of a rate tuple $b$. This rate tuple can be inserted, along with $H^*$, and the weight vector $w = [111...1]$, 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-channel’s GDFE

A GDFE with precoder for the BC can be constructed by a series of $U$ GDFE designs, without computation of the worst-case noise. The precoder is constructed by extracting user $u$’s rows only from the $u^{th}$ GDFE design’s unbiased feedback matrix $G_u^{unbiased}$. For receiver $u$ with white noise (so whitening absorbed into the channel definition):

$$y_u = H_u \left( \sum_{i=1}^{U} x_i \right) + n_u$$

$$= H_u [1 \ 1 \ \ldots \ 1] x + n_u$$

$$= P_u x + n_u .$$

(14.74)

(14.75)

(14.76)

The matrix $P_u$ adjusts (replicates) the channel matrix $H_u$ to correspond to each of the input user components. More generally, each of the input components may be related to its white unit-energy/dimension (innovations) data input as

$$x_u = A_u v_u ,$$

(14.77)

or thus effectively in the most general case, the channel matrix formed is

$$P_u = H_u [A_1 \ A_2 \ \ldots \ A_u] ,$$

(14.78)

The index of 1 has now tacitly been placed at the top since the order of the dual BC is reversed with respect to the dual MAC. The GDFE for receiver $u$ is then formed from

$$R_{b,u}^{-1} = P_u^* P_u + I = G_u^* S_{0,u,0} G_u ,$$

(14.79)

with unbiased feedback section

$$G_u^{unbiased} = I + S_{0,u} (S_{0,u} - I)^{-1} \cdot (G_u - I) .$$

(14.80)

The corresponding feedforward section is

$$W_u^{unbiased} = (S_{0,u} - I)^{-1} \cdot G_u^{-*} P_u^* R_{mm}^{-1/2}(u)$$

(14.81)

with inclusion of the noise-whitening filter shown explicitly.

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

```matlab
>> 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

>> Ht1=P1*[eye(3) eye(3)]
2.3505 -2.3505 0 2.3505 -2.3505 0
```
>> Ht2=P2*[eye(3) eye(3)] =

2.3505 2.1155 0 2.3505 2.1155 0
0 2.3505 2.1155 0 2.3505 2.1155

>> Rf1=Ht1'*Ht1 =

5.5249 -5.5249 0 5.5249 -5.5249 0
-5.5249 11.0497 -5.5249 -5.5249 11.0497 -5.5249
0 -5.5249 5.5249 0 -5.5249 5.5249
5.5249 -5.5249 0 5.5249 -5.5249 0
-5.5249 11.0497 -5.5249 -5.5249 11.0497 -5.5249
0 -5.5249 5.5249 0 -5.5249 5.5249

>> G1=chol(Rb1inv) =

2.5544 -2.1629 0 2.1629 -2.1629 0
0 2.7151 -2.0349 -0.4110 2.3468 -2.0349
0 0 1.5441 -0.4854 0.8964
0 0 0 1.2572 -0.2501 -0.2117
0 0 0 0 1.2515 -0.2935
0 0 0 0 0 1.2040

>> G1bar=diag(diag(G1))=

2.5544 0 0 0 0 0
0 2.7151 0 0 0 0
0 0 1.5441 0 0 0
0 0 0 1.2572 0 0
0 0 0 0 1.2515 0
0 0 0 0 0 1.2040

>> S01=G1bar*G1bar =

6.5249 0 0 0 0 0
0 7.3716 0 0 0 0
0 0 2.3841 0 0 0
0 0 0 1.5806 0 0
0 0 0 0 1.5662 0
0 0 0 0 0 1.4496

>> G1=inv(G1bar)*G1 =

1.0000 -0.8467 0 0.8467 -0.8467 0
0 1.0000 -0.7495 -0.1149 0.8643 -0.7495
0 0 1.0000 -0.2662 -0.3144 0.5806
0 0 0 1.0000 -0.1989 -0.1684
0 0 0 0 1.0000 -0.2345
0 0 0 0 0 1.0000

>> Gunbias1=S01*inv(S01-eye(6))*(G1-eye(6)) + eye(6) =
upper 3 rows are feedback for user 1 on user 1 and on user 2
ignore bottom 3 rows

>> Wunbias1=inv(S01-eye(6))*inv(G1')*Ht1' =

0.4254  0
-0.0565  0.3689
-0.1951  -0.4254
0.4254  0.1951
-0.1494  0.2760
-0.2035  -0.4254

last 3 rows need not be implemented at receiver 1

>> Rf2=Ht2'*Ht2 =

5.5249  4.9724  0  5.5249  4.9724  0
4.9724  10.0000  4.9724  4.9724  10.0000  4.9724
0  4.9724  4.4751  0  4.9724  4.4751
5.5249  4.9724  0  5.5249  4.9724  0
4.9724  10.0000  4.9724  4.9724  10.0000  4.9724
0  4.9724  4.4751  0  4.9724  4.4751

>> Rb2inv=Rf2+eye(6);
>> G2=chol(Rb2inv)=

2.5544  1.9466  0  2.1629  1.9466  0
0  2.6853  1.8517  0.2838  2.3129  1.8517
0  0  1.4305  -0.3674  0.4821  0.7314
0  0  0  1.2772  0.2214  -0.2011
0  0  0  0  1.2569  0.3035
0  0  0  0  0  1.1742

>> G2bar=diag(diag(G2))=

2.5544  0  0  0  0  0
0  2.6853  0  0  0  0
0  0  1.4305  0  0  0
0  0  0  1.2772  0  0
0  0  0  0  1.2569  0
0  0  0  0  0  1.1742

>> S02=G2bar*G2bar =

6.5249  0  0  0  0  0
0  7.2107  0  0  0  0
0  0  2.0463  0  0  0
\[
\begin{bmatrix}
0 & 0 & 0 & 1.6312 & 0 & 0 \\
0 & 0 & 0 & 0 & 1.5799 & 0 \\
0 & 0 & 0 & 0 & 0 & 1.3788
\end{bmatrix}
\]

\[G2=\text{inv}(G2\text{bar})*G2 =
\]
\[
\begin{bmatrix}
1.0000 & 0.7621 & 0 & 0.8467 & 0.7621 & 0 \\
0 & 1.0000 & 0.6896 & 0.1057 & 0.8613 & 0.6896 \\
0 & 0 & 1.0000 & -0.2568 & 0.3370 & 0.5113 \\
0 & 0 & 0 & 1.0000 & 0.1734 & -0.1574 \\
0 & 0 & 0 & 0 & 1.0000 & 0.2415 \\
0 & 0 & 0 & 0 & 0 & 1.0000
\end{bmatrix}
\]

\[\text{lower 3 rows are the feedback for user 2 on user 2, ignore top 3 rows}\]

\[\text{G_{unbias2}}=S02*\text{inv}(S02-\text{eye}(6))*(G2-\text{eye}(6)) + \text{eye}(6) =
\]
\[
\begin{bmatrix}
1.0000 & 0.9000 & 0 & 1.0000 & 0.9000 & 0 \\
0 & 1.0000 & 0.8006 & 0.1227 & 1.0000 & 0.8006 \\
0 & 0 & 1.0000 & -0.5023 & 0.6591 & 1.0000 \\
0 & 0 & 0 & 1.0000 & 0.4480 & -0.4068 \\
0 & 0 & 0 & 0 & 1.0000 & 0.6579 \\
0 & 0 & 0 & 0 & 0 & 1.0000
\end{bmatrix}
\]

\[\text{G_{unbias2}}=S02*\text{inv}(S02-\text{eye}(6))*(G2-\text{eye}(6)) + \text{eye}(6) =
\]

\[\text{W_{unbias2}}=\text{inv}(S02-\text{eye}(6))*\text{inv}(G2')*Ht2' =
\]
\[
\begin{bmatrix}
0.4254 & 0 \\
0.0522 & 0.3785 \\
-0.2137 & 0.4727 \\
0.4254 & -0.1923 \\
0.1272 & 0.3110 \\
-0.2239 & 0.4727
\end{bmatrix}
\]

\[\text{top 3 rows need not be implemented at receiver 2}
\]
\[\text{(design separate GDFE for each user at receiver using Rxx(u), Hu, and earlier-user noise)}\]

The example illustrates the appropriate setting of the $R_{xx}(u)$ at the output of the transmit filter matrix $A$. This process basically absorbs the $A_u$ into the feedback coefficients between different users. Since the point fed back is $x_n$, 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 $R_{xx}^u(u)$ used to shape the channel input, which also must then be used to filter the feedback subtractions since this alters the channel by this same amount.
Figure 14.14: GDFE BC transmitter adjustment for matching dual energies.
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. 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. 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. Section 4.8 also discusses the Zipper method by Isaksson that can be used when a downstream BC or an upstream MAC share the same users in a bi-directional multi-user channel. Such alignment also 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” cancellers (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 received 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 \( \tilde{H}_u \) (that is \( \nu T' \geq \max_{u,i} \{ \max_{t} |\tilde{h}_{u,i}(t)| \} \)), 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

\[
Y_n = \underbrace{H_n}_{L_y U \times 1} \cdot \underbrace{X_n}_{L_x \times 1} + \underbrace{N_n}_{L_y U \times 1},
\]

(14.82)

where

\[
H_n = \begin{bmatrix} H_{1,n} \\ \vdots \\ H_{U,n} \end{bmatrix}
\]

(14.83)
Figure 14.16: Illustration of the Vector DMT BC system.

\[
X_n = \begin{bmatrix} x_{1,n} \\ \vdots \\ x_{L_x,n} \end{bmatrix} = \sum_{u=1}^{U} X_{u,n} \quad (14.84)
\]

\[
Y_n = \begin{bmatrix} Y_{1,n} \\ \vdots \\ Y_{U,n} \end{bmatrix} \quad (14.85)
\]

\[
Y_{u,n} = \begin{bmatrix} Y_{u,1,n} \\ \vdots \\ Y_{u,L_y,n} \end{bmatrix} = P_{u,n} V_n + N_{u,n} \quad (14.86)
\]

where \( P_{u,n} = H_{u,n} [A_{1,n} A_{2,n} \ldots A_{U,n}] \) and \( V_n \) is the input vector of unit(identity)-energy user inputs on tone \( n \), with user 1 at the top in keeping with our BC duality order-reversal. The quantities \( A_{u,n} \) simply form the individual user autocorrelation matrix components (or energy scalings in the scalar case). 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

\[
\sum_n \text{trace}\{R_{XX}(n)\} \leq E_x. \quad (14.88)
\]

The input autocorrelation on tone \( n \) is

\[
R_{XX}(n) = E[X_n X_n^*] = \sum_{u=1}^{U} R_{XX}(u,n) \quad (14.89)
\]

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

\[
R_{XX}(n) = \sum_{u=1}^{U} R_{XX}(u,n) = \sum_{u=1}^{U} A_{u,n} \cdot R_{V^*V^*}(u,n) \cdot A_{u,n}^*. \quad (14.90)
\]
Receiver $u$’s noise (on any tone) can be found then as

$$
\tilde{R}_{\text{noise}}(u,n) = R_{NN}(u,n) + \sum_{i=u+1}^{U} H_{u,n} \cdot R_{XX}(i,n) \cdot H_{u,n}^*.
$$  \hspace{1cm} (14.91)

The noise-equivalent channel is then

$$
\bar{H}_{u,n} = \frac{1}{2} \tilde{R}_{\text{noise}}^{-1/2}(u,n) \cdot H_{u,n} \cdot A_{u,n}.
$$  \hspace{1cm} (14.92)

More directly, a GDFE for receiver $u$ will estimate user $u$’s signal with users $i = u + 1, \ldots, U$ removed and correspond to $I(x_u; Y_{u,n}/x_{u+1} \ldots x_U)$. Then a series of such designs on each tone is done for $u = 1, \ldots, U$ and the relevant rows of each feedback section $G_{u,n}^{\text{unbiased}}$ forms the precoder for each tone.

### 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 $16 \times 2.208$ 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

\(\text{VDSL2 actually doubles carrier frequency width and leaves the number of tones at 4096 to cover exactly the same bandwidth, but the program used here was based on 4.3125 kHz spacing.} \)
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.

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 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 14.19: 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 $b'$ with $b_1' = 0, \ldots, b_{1,\text{max}}$, ... $b_U' = 0, \ldots, b_{U,\text{max}}$ with increments selected appropriately and maxima 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 $\mathcal{E}_{\text{vec}}$ for a given $b$ on the dual MAC using the minPmac program of Section 13.5.
   
   (b) if $\sum_u E_u \leq \mathcal{E}_{\text{vec}}$, then the point is in the region, so $c_{\text{new}}(b) = \{b' \cup c_{\text{old}}(b)\}$.

3. Trace the boundary for all points in which $\sum_u E_u = \mathcal{E}_{\text{vec}}$. 

479