STANFORD

Lecture 8

Multiple Access Channels

April 26, 2023

JOHN M. CIOFFI

Hitachi Professor Emeritus of Engineering

Instructor EE392AA – Spring 2023

Announcements & Agenda

- Announcements
  - Problem Set #4 is due Tuesday (no late)
  - PS1 - 2 solutions are at the site, PS4 at 17:05 Tuesday
    - PS3 as soon as last homework in
  - Midterm is 5/3 in class
  - May 19; 3pm make-up class (replaces May 15) location changed to 200-003
  - Office hours today 4:30-5:00, then probably 5:30-6:00

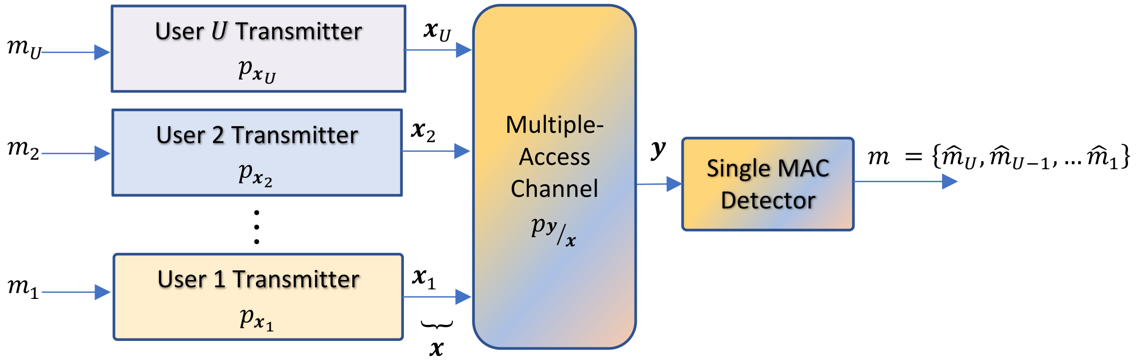
- Agenda
  - MAC $\mathcal{C}(\boldsymbol{b})$ via partial rate sums
  - Scalar Gaussian MAC
  - Vector Gaussian MAC
    - mu_mac.m software
  - Capacity Region for frequency-indexed channels

MAC $\mathcal{C}(\boldsymbol{b})$ via partial rate sums

PS4.3 - 2.23

The MAC's partial rate sums

$p_{\boldsymbol{x}} = \prod_{u=1}^{U} p_{\boldsymbol{x}_u}$ independent user inputs



- User u has maximum bit rate, when all other users are given (cancelled): $\boxed{b_u \leq \mathcal{I}(\boldsymbol{x}_u; \boldsymbol{y}\,/\boldsymbol{x}_{U\setminus u})}$
- The single receiver can process user subset $\boldsymbol{u} \subseteq \boldsymbol{U}$
  - has single-macro-user interpretation with summed bits/subsymbol:
  - $b_{\boldsymbol{u}} = \sum_{u \in \boldsymbol{u}} b_u \leq \mathcal{I}(\boldsymbol{x}_{\boldsymbol{u}}; \boldsymbol{y}\,/\boldsymbol{x}_{U\setminus \boldsymbol{u}})$
  - this is a hyperplane with $|\boldsymbol{u}| - 1$ dimensions ($\in \mathbb{C}^{|\boldsymbol{u}|}$)
- Order simplifies (receiver) to $\boldsymbol{\Pi} = \boldsymbol{\pi}_1$; The user order within $\boldsymbol{u}$ does not change the sum $\mathcal{I}(\boldsymbol{x}_{\boldsymbol{u}}; \boldsymbol{y}\,/\boldsymbol{x}_{U\setminus \boldsymbol{u}})$, nor does the order within $\boldsymbol{U} \setminus \boldsymbol{u}$.
  - The number of planes (lines … hyperplanes) to search decreases substantially to $2^U - 1$ (null set excluded) << $(U!)^U$ (large U).

Chain-Rule Reminder Lemma 2.3.4

$$\boxed{\mathcal{I}(\boldsymbol{x};\boldsymbol{y}) = \mathcal{I}\left(\boldsymbol{x_u};\boldsymbol{y}\,/\boldsymbol{x}_{\boldsymbol{U}\backslash\boldsymbol{u}}\right)+\mathcal{I}\left(\boldsymbol{x}_{\boldsymbol{U}\backslash\boldsymbol{u}}\,;\boldsymbol{y}\,\right)}$$

User (set) $\boldsymbol{u}$ detected with all other users $\boldsymbol{x}_{\boldsymbol{U}\backslash\boldsymbol{u}}$ given (cancelled)

Other-user (set) $\boldsymbol{U}\backslash\boldsymbol{u}$ detected with users $\boldsymbol{x_u}$ as noise

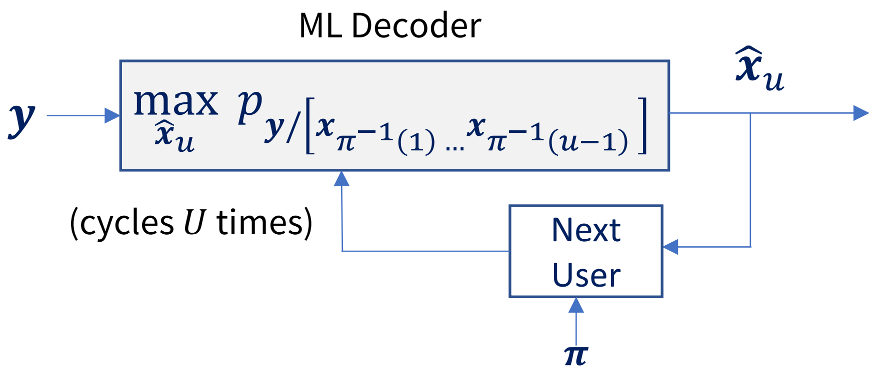
2^U possible choices of $\boldsymbol{u}$

- $b \leq \mathcal{I}(\boldsymbol{x};\boldsymbol{y})$ - the rate sum (corresponds to choice $\boldsymbol{u}=\boldsymbol{U}$)
- A (hyperplane) **face**: $b_1+b_2+\cdots+b_{|\boldsymbol{u}|} \leq \mathcal{I}\left(\boldsymbol{x_u};\boldsymbol{y}\,/\boldsymbol{x}_{\boldsymbol{U}\backslash\boldsymbol{u}}\right)$ - partial rate sums ($2^{|\boldsymbol{u}|}$ -1)
  - There are also U trivial faces for positive bits/subsymbol $b_u \geq 0$, so really 2^u -1+U faces that bound $\mathcal{A}(\boldsymbol{b}, p_{\boldsymbol{x}})$.
- A **vertex** corresponds to a specific $\boldsymbol{b} = \mathcal{I}$ for a specific order $\boldsymbol{\pi}$.

$$\begin{bmatrix}\mathcal{I}(\boldsymbol{x}_2;\boldsymbol{y}\,/\boldsymbol{x}_1)\\ \mathcal{I}(\boldsymbol{x}_1;\boldsymbol{y}\,)\end{bmatrix} \quad \begin{bmatrix}\mathcal{I}(\boldsymbol{x}_1;\boldsymbol{y}\,/\boldsymbol{x}_2)\\ \mathcal{I}(\boldsymbol{x}_2;\boldsymbol{y})\end{bmatrix}$$

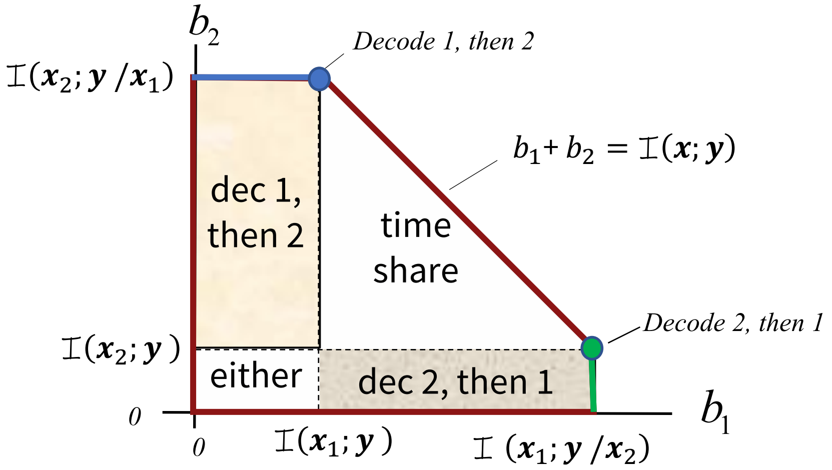
When $U = 2$; in general, $U!$ vertices for a specific $p_{\boldsymbol{x}}$

Chain-Rule Decoder



Successive Decoding or …
Generalized Decision Feedback Eq

- For the given order, decode all the lower-indexed users first and then current user
- Since there is only one order, relabel users and avoid all the $\pi^{-1}(\cdot)$ notation
- No loss of generality

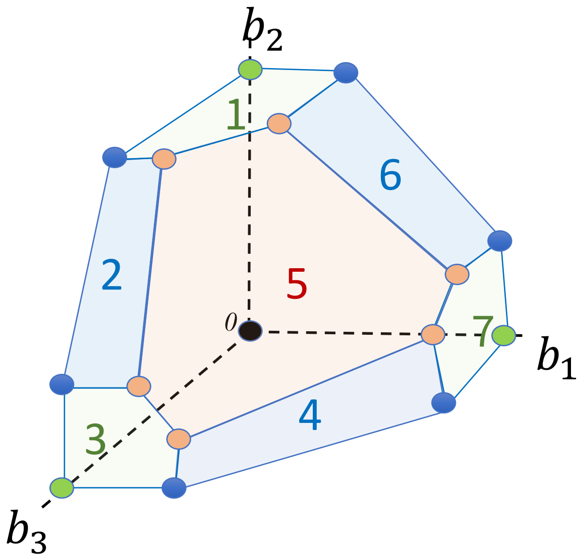


Specific to a $p_{\boldsymbol{xy}}$

- Pentagon – 5 vertices and 5 faces
  - $2^U - 1 + U$ Faces are the $\mathbb{I}\left(\boldsymbol{x}_{\boldsymbol{u}} ; \boldsymbol{y} / \boldsymbol{x}_{\boldsymbol{U} \setminus \boldsymbol{u}}\right)$ & $b_u \geq 0$
  - $U! = 2$ vertices are the both-user order points $\boldsymbol{\pi}$
    - 2 more are single-user points, one for each user
    - 1 more is the origin
    - 5 total

- b_2 vertex (and short blue line) decodes 1 first (given), then 2 as if 1 is "cancelled."
  - Similar statement for b_1 vertex (and short green) line
- Line with slope -1 is time-share or really vertex-share; it also is constant maximum rate sum (for this $p_{\boldsymbol{xy}}$)
  - There are two codes for each user (4 codes); example of user components (or subusers, sometimes called "rate splitting")

A 3-user rate region



- Decahedron – 10 faces
  - $2^U - 1 + U$ Faces are the $\mathbb{I}(\boldsymbol{x_u}\,; \boldsymbol{y}\,/\boldsymbol{x_{U\backslash u}})$
  - $U! = 6$ vertices (rose) are the 3-user order points $\boldsymbol{\pi}$

= vertex for 3 users $U!$ $(= 6)$
= vertex for 2 users $U!$ $(= 6)$
= vertex for 1 user U $(= 3)$
= vertex for 0 users $(= 1)$

$\sum_{k=0}^{U} \binom{U}{k} \cdot k! =$16 vertices

- b_2 horizontal plane (pentagon) decodes 1 and 3 first (given), then 2 as if 1 and/or 3 are "cancelled."
  - 1 and 3 form a two-user horizontal pentagon region
  - Similar statements for b_1 vertical-plane pentagon and b_3 facial-plane pentagon
- Rose plane normal to $\mathbf{1} = [1 \quad 1 \quad 1]^*$ is dimension-share of rose vertices; it has constant maximum rate sum (for this $p_{\boldsymbol{xy}}$)
  - Could be as many as 3 codes/components for each user on a time-share of vertices
- The blue and green planes may also dimension-share vertices
- $\mathcal{A}(\boldsymbol{b}, p_{\boldsymbol{x}})$ is the entire interior plus faces and vertices. Any point outside violates at least one single-user mutual-information bound.

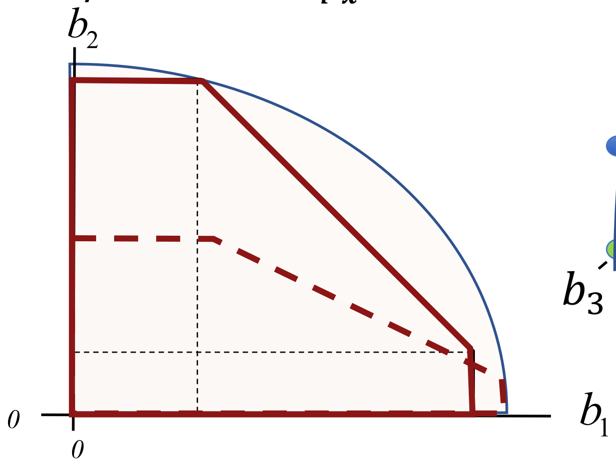
MAC Capacity Region

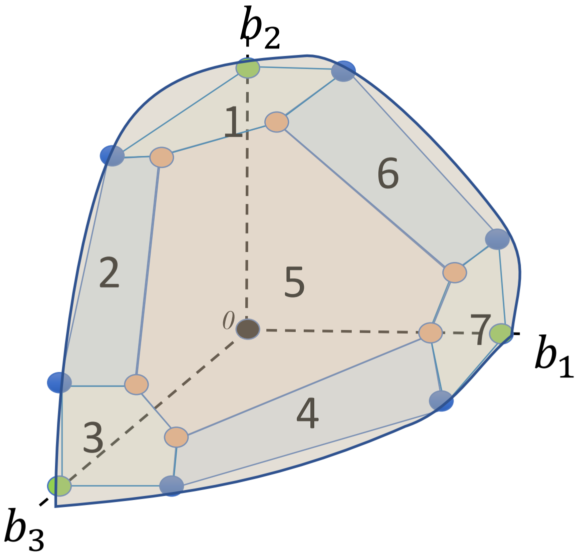
- More formally, the MAC's achievable region is bounded by hyperplanar regions

$$\mathcal{A}(\boldsymbol{b}, p_{\boldsymbol{x}}) = \bigcap_{\boldsymbol{u}\subseteq \boldsymbol{U}} \left\{ \boldsymbol{b} \,\middle|\, 0 \le \sum_{i\in\{\boldsymbol{u}\}} b_i \le \mathcal{I}\left(\boldsymbol{x}_i; \boldsymbol{y} / \boldsymbol{x}_{\boldsymbol{u}\setminus i}\right) \right\}$$

- The vertices are where hyperplanes intersect at a point
  - So lines (smaller dimensional hyperplanes) also bound …
- Convex hull over all multi-user input probability distributions $p_{\boldsymbol{x}}$

$$\mathcal{C}_{MAC}(\boldsymbol{b}) = \overset{conv}{\bigcup_{\boldsymbol{u}\subseteq p_{\boldsymbol{x}}}} \mathcal{A}(\boldsymbol{b}, p_{\boldsymbol{x}})$$



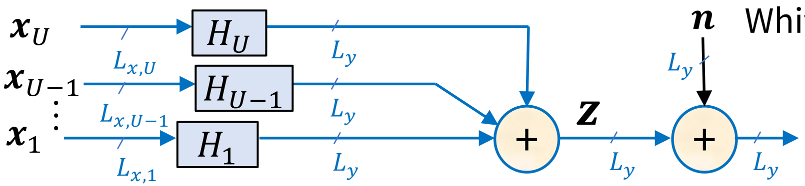


Scalar Gaussian MAC

PS4.3 - 2.25 Time-Division Multiplexing region

Section 2.7.2

General Gaussian MAC



White Gaussian Noise

$R_{\boldsymbol{nn}} = \sigma^2 \cdot I$

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

$$\underbrace{\boldsymbol{y}}_{L_y \times 1} = \underbrace{[H_U \quad H_{U-1} \quad \cdots \quad H_1]}_{\substack{H \\ L_y \times \mathfrak{L}_x}} \cdot \underbrace{\begin{bmatrix} \boldsymbol{x}_U \\ \boldsymbol{x}_{U-1} \\ \vdots \\ \boldsymbol{x}_1 \end{bmatrix}}_{\substack{\boldsymbol{x} \\ \mathfrak{L}_x \times 1}} + \boldsymbol{n}$$

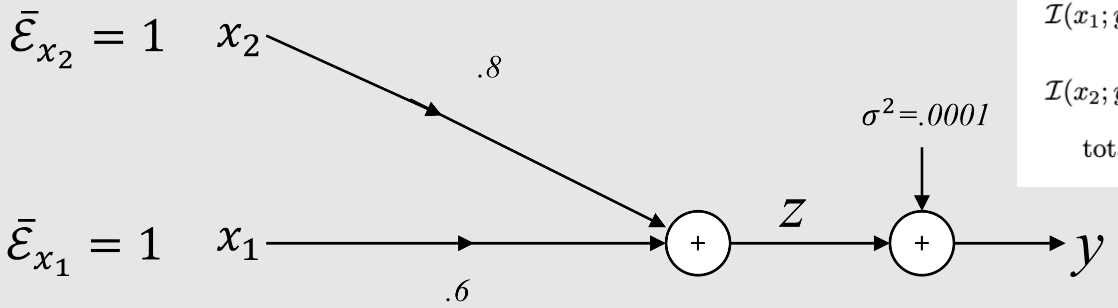
More generally, variable-dim inputs

$$\mathfrak{L}_x = \sum_{u=1}^{U} L_{x,u} \sim U \cdot L_x$$

- Inputs are independent
  - $R_{\boldsymbol{xx}}$ is block diagonal
  - Only 1 output and 1 noise
- One Receiver will estimate all inputs
  - Can do so in any order
  - "Given an input" $\boldsymbol{x}_u$ means cancel it from $\boldsymbol{y}$
  - This does not necessary mean subtract $H_u \cdot \boldsymbol{x}_u$ from $\boldsymbol{y}$
    - Unless $L_y = L_{x,u} = 1$; or H_u is diagonal and noise is white

ϱ_H is the matrix H's **rank**
number of linearly independent rows (or columns)
= # of non-zero singular values

Example

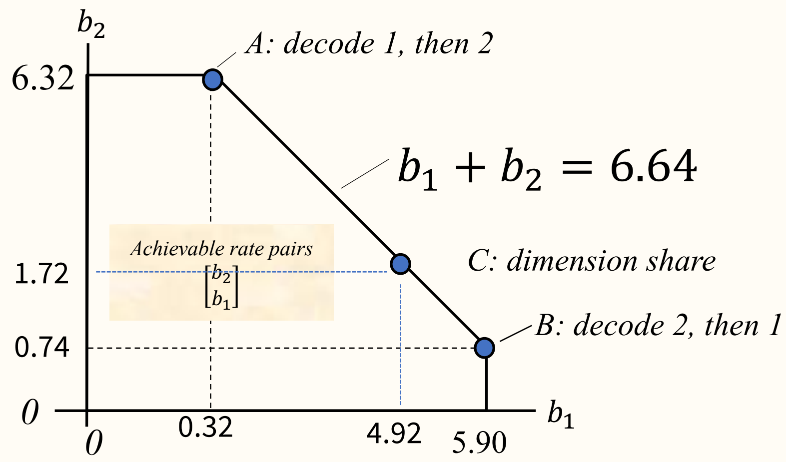


$$\mathcal{I}(x_1;y) = \frac{1}{2}\log_2\left(1+\frac{.36\cdot 1}{.0001+.64}\right) = .32 \text{ bits/dimension}$$
$$\mathcal{I}(x_2;y) = \frac{1}{2}\log_2\left(1+\frac{.64\cdot 1}{.0001+.36}\right) = .74 \text{ bits/dimension}$$
$$\text{total} = 1.06 \text{ bits/dimension} \quad ,$$

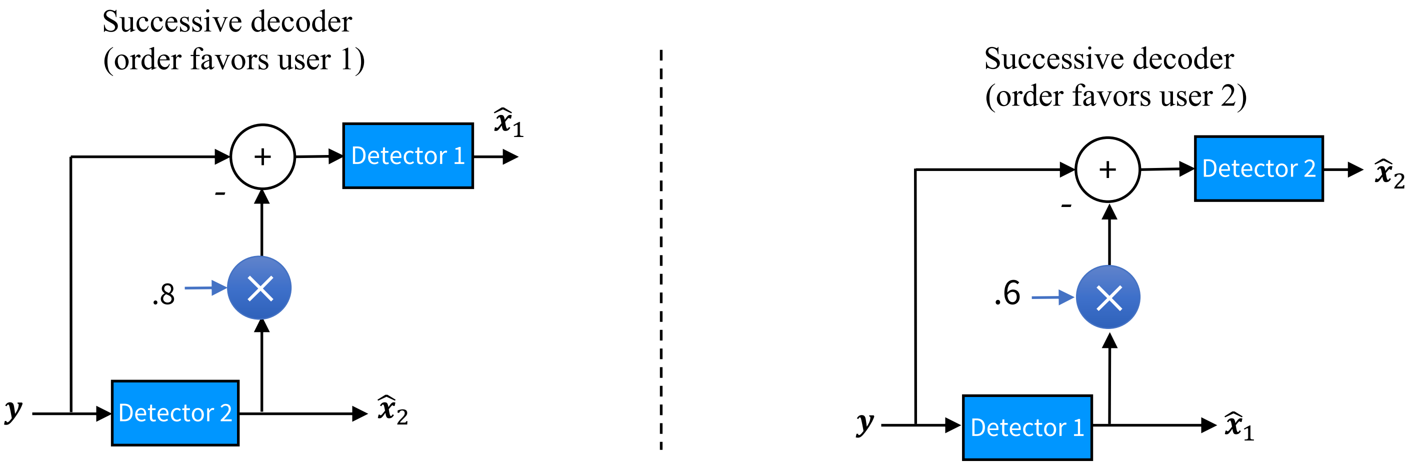
$$\mathcal{I}(x_2;y/x_1) = \frac{1}{2}\log_2(1+\text{SNR}_2) = \frac{1}{2}\log_2\left(1+\frac{.64\cdot 1}{.0001}\right) = 6.32 \text{ bits/dimension}$$

$$\mathcal{I}(x_1;y/x_2) = \frac{1}{2}\log_2(1+\text{SNR}_1) = \frac{1}{2}\log_2\left(1+\frac{.36\cdot 1}{.0001}\right) = 5.90 \text{ bits/dimension}$$

- Point C is ¼ share B and ¾ share A



Successive decoding for scalar example



- Only 2 orders possible for 2 users
- $U!$ in general (corresponding to each possible order)
- The last user is "favored" in decoding (first accepts other as noise)

2 – User Scalar $L_x = L_y = 1$

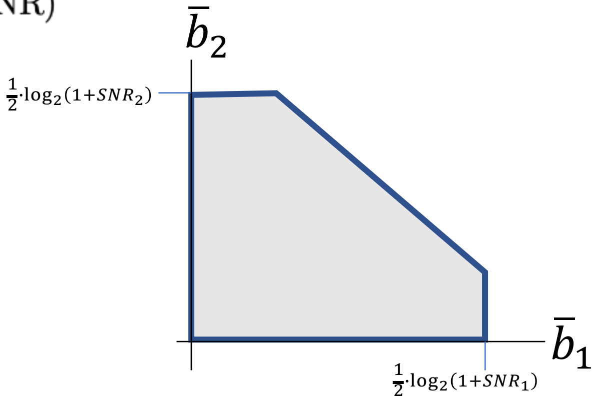
General formula
Scalar MAC

$$\bar{I}(x_{\boldsymbol{u}}; y/\boldsymbol{x}_{\boldsymbol{U}\setminus\boldsymbol{u}}) = \frac{1}{2}\log_2\left(1 + \frac{\sum_{i\in\boldsymbol{u}} \bar{\mathcal{E}}_i \cdot |H_i|^2}{\sigma^2}\right)$$

2 users

$$\begin{aligned} \mathrm{SNR}_1 &= \frac{\mathcal{E}_1 \cdot |h_1|^2}{\sigma^2} \\ \mathrm{SNR}_2 &= \frac{\mathcal{E}_2 \cdot |h_2|^2}{\sigma^2} \\ \mathrm{SNR} &= \frac{\mathcal{E}_1 \cdot |h_1|^2 + \mathcal{E}_2 \cdot |h_2|^2}{\sigma^2} \end{aligned}$$

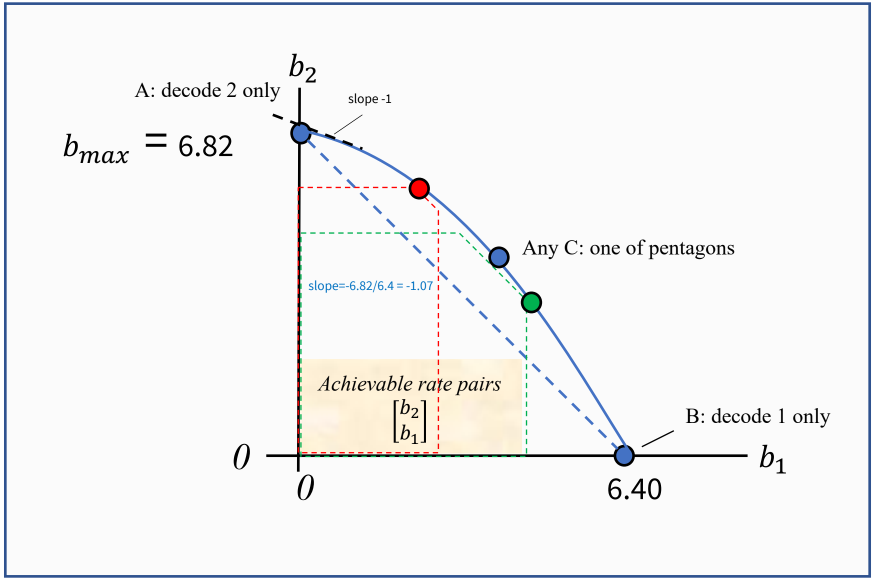
$$\begin{aligned} \bar{b}_1 &\le \frac{1}{2}\log_2(1+\mathrm{SNR}_1) \\ \bar{b}_2 &\le \frac{1}{2}\log_2(1+\mathrm{SNR}_2) \\ \bar{b}_1 + \bar{b}_2 &\le \frac{1}{2}\log_2(1+\mathrm{SNR}) \end{aligned}$$



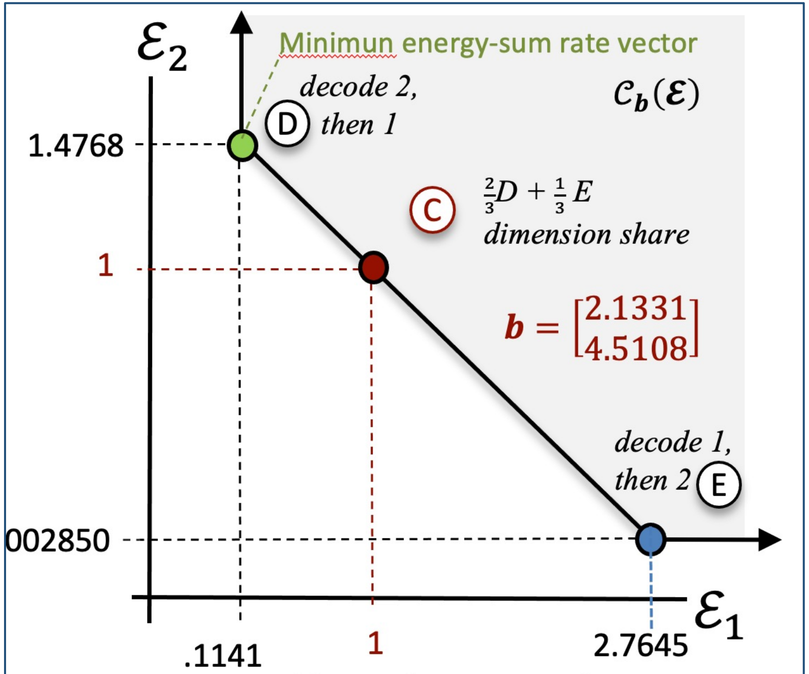
- 3 planes (lines) ~ 3 SNR's
- 1 sum rate
- Nonzero individual rates

Energy-Sum MAC

- Single energy constraint $\mathcal{E}_1 + \mathcal{E}_2 \le \mathcal{E}_x$ (instead of 2 constraints)
- Capacity region becomes union of pentagons (and 1 triangle),
  - one for each combination of energies that add to total.



- Or view Energy-Capacity Region
  - one for each bit vector $\boldsymbol{b}$



b). capacity energy region

Time-Sharing Conundrum

- What is meaning of time-sharing? ("convex hull")
  - The different codes correspond to user components, each used for its respective fraction of "time" (dimensions).

- With time-sharing, what does $\mathcal{E}_u$ mean?
  - Energy constant at $\mathcal{E}_u$: Is this then for every symbol/subsymbol in the sharing?
  - Or the average over the "time-shared" subsymbols?

- The second instance of averaging often enlarges the capacity region

- So "time-sharing" is somewhat ill-defined
  - Despite most info/com texts on MAC using it

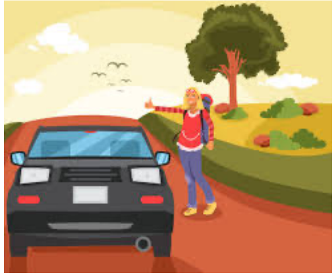
- Lecture 4's Separation Theorem actually allows different mutual information $\mathcal{I}_A$ and $\mathcal{I}_B$ to be represented by their average information – *for the same user*
  - $\mathcal{I} = \alpha \cdot \mathcal{I}_A + (1-\alpha) \cdot \mathcal{I}_B$
  - ST uses same constellation with average $\boldsymbol{b}$ for each symbol, possible very large$|C|$).
  - If the shared same-user codes correspond to vertices with different orders, this creates issues for Separation Thm application.
    - But it is still possible, although the successive decoding needs to become "iterative-user" successive decoding.
    - Of course each user can use subusers; each user has subcode for A and for B, but constellation varies.

Primary and Secondary Components (E-sum MAC)

Primary-user component: has nonzero energy for E-sum MAC's maximum rate.

Secondary-user component: has zero energy for E-sum MAC's maximum rate.

- Primary components dominate with largest pass-space gains (dimensions used for component).
- Secondary users "free load" on these primary-component dimensions.



Previous example (.8 and .6):

The pass-space is just one dimension ($L_y = 1$).

user 2 is all primary (.8) ; user 1 is all secondary (.6).

max sum is 6.82 (all energy on user 2).

Rate-sum decreases if secondary user components energize (see slide 15).

How Many Primary Components (E-sum MAC)?

- The MAC has no more than $U^o \leq \wp_H$ primary components, to find them first do U SVD's:

$$\tilde{H}_u = R_{noise}^{-1/2}(u) \cdot H_u = F_u \cdot \Lambda_u \cdot M_u^* \quad \text{with} \quad \left|\tilde{H}_u\right| \triangleq \prod_{l=1}^{\wp_{H_u}} \lambda_{u,l} > 0\,.$$

- Each user can excite up to $\wp_{H_u}$ possible independent dimensions per subsymbol
  - The $R_{noise}(u)$ includes all other user components' crosstalk for whatever energies they use (knows all $R_{\boldsymbol{xx}}(u)$'s).
  - Each user can have vector-coding modulator without loss, or some linear combination of the pass-space dimensions.

- For the channel gains in the VC,

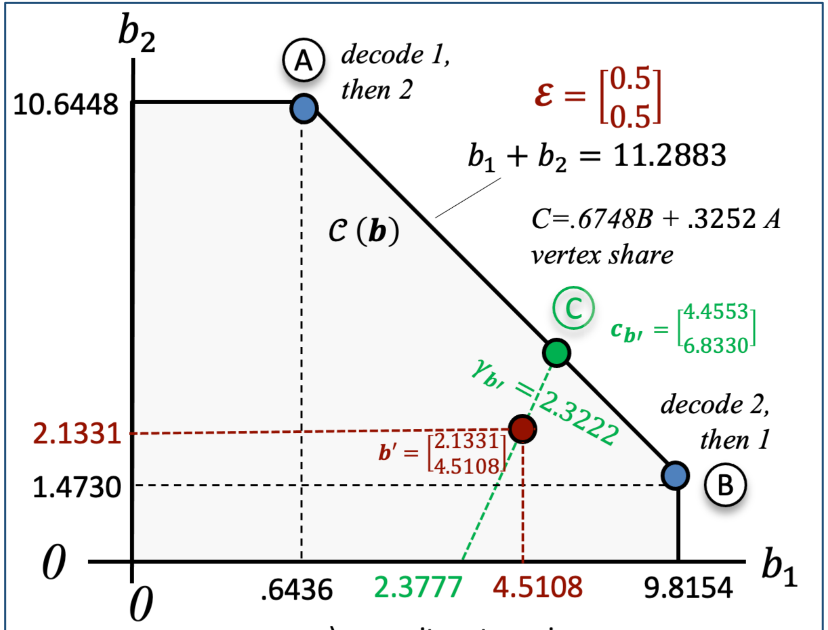
$$g_u = \left|\tilde{H}_u\right|^2 = \prod_{l=1}^{\wp_{H_u}} \lambda_{u,l} \quad .$$

- The **primary-user components** correspond to those energized in achieving max rate sum on the E-sum MAC. All others are **secondary-user components**.

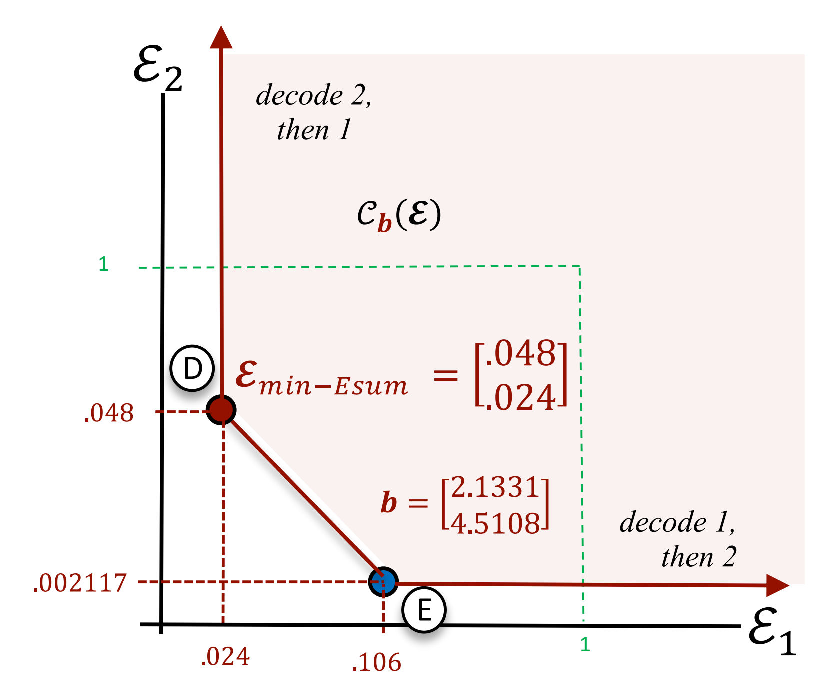
- The "components" idea is helpful when individual users' transmitters have >1 dimension (MIMO), via
  - time-sharing, DMT, and/or multiple antennas

Conundrum: double-sampling-rate Example

[80 60] channel again at twice sampling rate



a). capacity rate region



b). capacity energy region

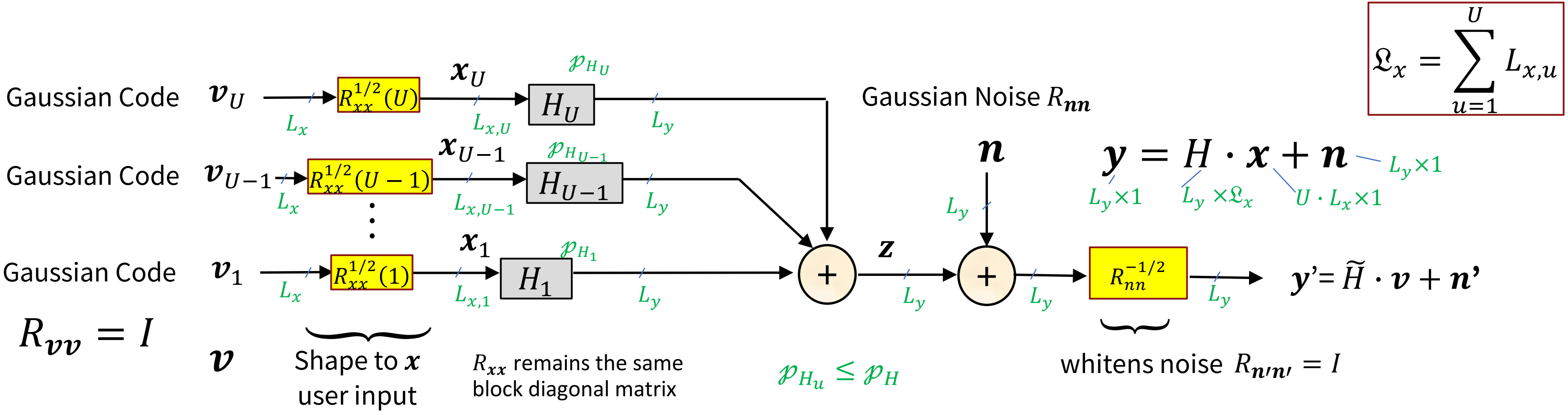
- The vector $\boldsymbol{b}$ is now in the interior of the region, although is it the same channel?
  - The time-sharing needs to occur at the same sampling rate, meaning the symbol period increases, for the original $\mathcal{C}(\boldsymbol{b})$ **to remain correct**

Vector Gaussian MAC

PS4.4 - 2.24 MAC regions

Section 2.7.2.2

MAC ~ single channel with white input



- Normalizes (redefines, not $R_{noise}(u)$ here) individual user MAC channels to $\tilde{H}_u \triangleq R_{\boldsymbol{nn}}^{-1/2} \cdot H_u \cdot R_{\boldsymbol{xx}}^{1/2}(u)$
- **Normalized MAC** is now $\boldsymbol{y}' = \tilde{H} \cdot \boldsymbol{v} + \boldsymbol{n}'$ where
  - New input(s) is (are) "white" , $R_{\boldsymbol{vv}} = I$.
  - New noise is "white" , $R_{\boldsymbol{n'n'}} = I$
  - Drop the primes going forward $\boldsymbol{y} = \tilde{H} \cdot \boldsymbol{v} + \boldsymbol{n}$ → $\tilde{H}$ ' s dimensions carry the information (secondary may freeload)

Cholesky Factorization

- Related to MMSE linear-prediction (see Appendix D)

- Positive definite Hermitian symmetric $R = G^* \cdot S \cdot G$
  - G is upper triangular monic (1's on diagonal)
  - S is positive real diagonal matrix (even if R is complex)

- Matlab command is "**chol**" for *lower* $\times$ *upper* (lower is upper*) – **produces upper**
  - The matlab default command produces also upper, but is lower x upper (which is not the one above)
  - Gtemp=chol (R)
  - G= inv(diag(diag(Gtemp)))*Gtemp
  - S= diag(diag(Gtemp))*diag(diag(Gtemp))

- The following will use this factorization

Forward and Backward Canonical Channels

- Forward Canonical Channel
  - Output of matched-filter matrix

$$\boldsymbol{y}' = \underbrace{\tilde{H}^* \cdot \tilde{H}}_{R_f} \cdot \boldsymbol{\nu} + \underbrace{\tilde{H}^* \cdot \boldsymbol{n}}_{\boldsymbol{n}'}$$

- **MMSE Estimator** for backward channel

$$R_{\boldsymbol{\nu}\boldsymbol{y}'} \cdot R_{\boldsymbol{y}'\boldsymbol{y}'}^{-1} = R_f \cdot [R_f \cdot R_f + R_f]^{-1} = [R_f + I]^{-1} = R_b$$

- **Backward Canonical** Channel

$$\boldsymbol{\nu} = R_b \cdot \boldsymbol{y}' + \boldsymbol{e}$$

- Use **Cholesky** on backward-channel inverse $R_b^{-1} \triangleq R_f + I = G^* \cdot S_0 \cdot G$

$$\boldsymbol{y}'' = S_0^{-1} \cdot G^{-*} \boldsymbol{y}' \qquad \text{(algebra)}$$

$$\boxed{\boldsymbol{y}'' = G \cdot \boldsymbol{\nu} - \boldsymbol{e}'} \qquad \text{where} \qquad R_{\boldsymbol{e}'\boldsymbol{e}'} = S_0^{-1}$$

Back Substitution

- Not ML/MAP, but **successive decoding**
  - And **canonical – achieves capacity reliably, each user**
  - **if decisions are correct**
  - If $\Gamma > 0$ dB, then iterative decoding that approx. ML may be needed
- Each of these is MMSE based,
  - which we know is related to conditional $\mathcal{I}$.
- Much simpler decoder ("GDFE")
- SNR (biased) for each decision/dimension is $S_{0,u,l}$.
- But also

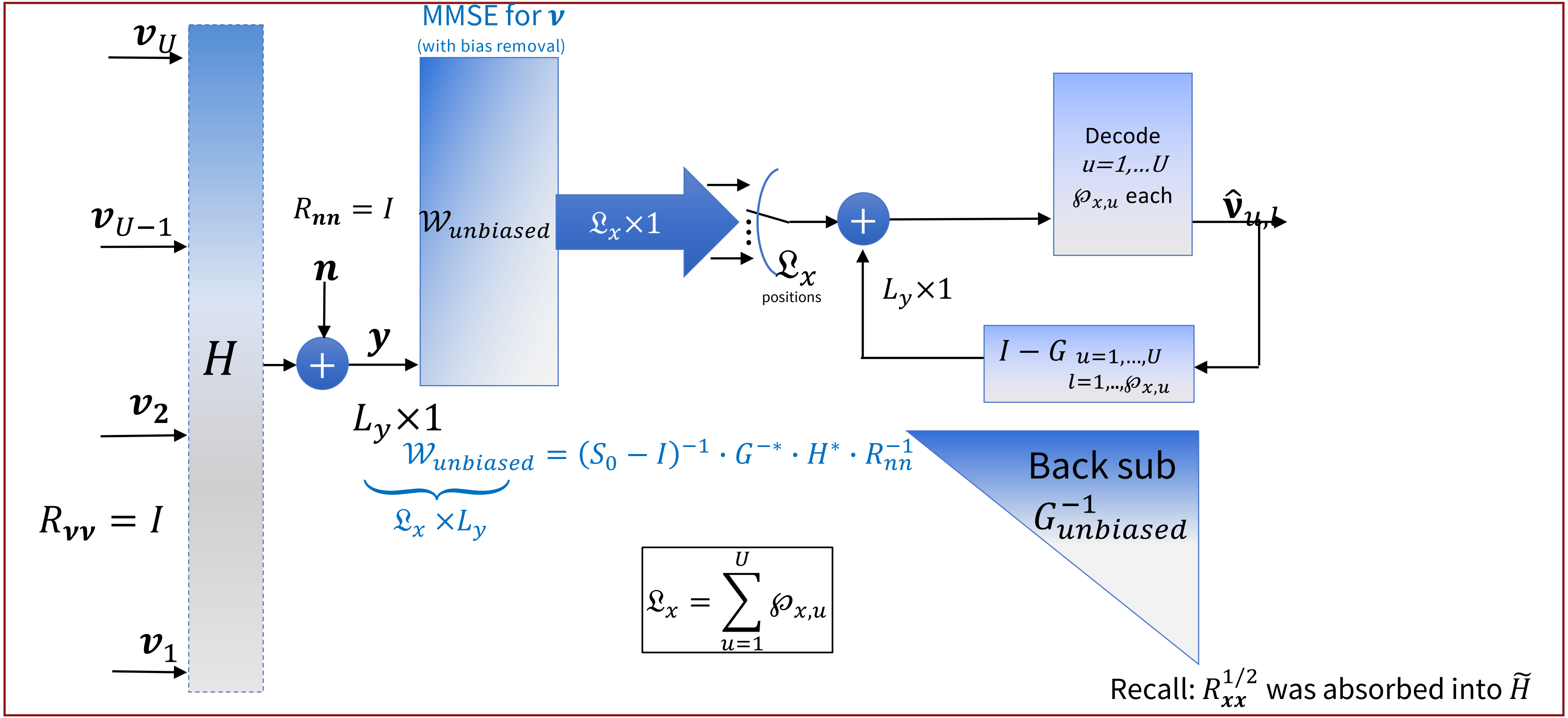
$$G = \begin{bmatrix} 1 & g_{U,U-1} & \cdots & g_{U,2} & g_{U,1} \\ 0 & 1 & \cdots & g_{U-1,2} & g_{U-1,1} \\ \vdots & \ddots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & g_{2,1} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} \hat{\boldsymbol{\nu}}_1 &= \text{decision}\left(\boldsymbol{y}_1''\right) \\ \hat{\boldsymbol{\nu}}_2 &= \text{decision}\left(\boldsymbol{y}_2'' - g_{2,1} \cdot \hat{\boldsymbol{\nu}}_1\right) \\ \vdots \quad & \vdots \quad \vdots \\ \hat{\boldsymbol{\nu}}_{U'} &= \text{decision}\left(\boldsymbol{y}_U'' - \sum_{i=1}^{U-1} g_{U,i} \cdot \hat{\boldsymbol{\nu}}_i\right) \end{aligned}$$

$$\mathcal{I}(\boldsymbol{x};\boldsymbol{y}) = \log_2(|\underbrace{\tilde{H}^*\tilde{H} + I}_{R_b^{-1}}|) = \log_2 | S_0 | = \log_2 \left\{ \prod_{u=1}^{U'} \prod_{\ell=1}^{L_{x,u}} SNR_{mmse,u,\ell} \right\} \text{bits / complex symbol} \ .$$

New parallel "independent" subchannels

Vector MAC Receiver



- Each user/decoder achieves $\mathcal{I}(\boldsymbol{v}_u; \boldsymbol{y} / [\boldsymbol{v}_{u-1} \ \cdots \ \boldsymbol{v}_1])$

Example 1

$$\tilde{H} = \begin{bmatrix} 5 & 2 \\ 3 & 1 \end{bmatrix}$$

```
>> Rf=H'*H =
   34    13
   13     5
>> Rbinv=Rf+eye(2) =
   35    13
   13     6

>> Gbar=chol(Rbinv)   =
   5.9161    2.1974
        0    1.0823

>> S0=diag(diag(Gbar))*diag(diag(Gbar)) =
  35.0000         0
        0    1.1714
>> G = inv(diag(diag(Gbar)))*Gbar =
   1.0000    0.3714
        0    1.0000
>> >> b=0.5*log2(diag(S0))  =
   2.5646
   0.1141
>> sum(b) =    2.6788
```

REVERSE ORDER – same commands –other vertex

```
>> H=[   2    5
   1    3] ;
Rbinv =
    6    13
   13    35
Gbar =
 2.4495    5.3072
      0    2.6141
S0 =
   6.0000        0
        0   6.8333
G =
 1.0000    2.1667
      0    1.0000
b =
 1.2925
 1.3863

sum(b) =   2.6788
```

- These are the two vertices for time-share

Example 1 continued

- Receiver filters and bias

Vertex 1

```
>> W=inv(S0)*inv(G') =
    0.0286         0
   -0.3171    0.8537
>> Wunb=S0*inv(S0-eye(2))*W =
    0.0294         0
   -2.1667    5.8333

>> MSWMFu=Wunb*H' =
    0.1471    0.0882
    0.8333   -0.6667
>> Gunb=eye(2)+S0*inv(S0-eye(2))*(G-eye(2)) =
    1.0000    0.3824
         0    1.0000

>> MSWMFu*H =
    1.0000    0.3824
    2.1667    1.0000
```

Vertex 2

```
>> W=inv(S0)*inv(G') =
    0.1667         0
   -0.3171    0.1463
>> Wunb=S0*inv(S0-eye(2))*W =
    0.2000         0
   -0.3714    0.1714
>> MSWMFu=Wunb*H'  =
    0.4000    0.2000
    0.1143    0.1429
>> Gunb=eye(2)+S0*inv(S0-eye(2))*(G-eye(2)) =
    1.0000    2.6000
         0    1.0000
>> MSWMFu*H=
    1.0000    2.6000
    0.3714    1.0000
```

- Not really triangular, why?

Easier with mu_mac.m

```
function [b, GU, WU, S0, MSWMFU] = mu_mac(H, A, Lxu, cb)

%
[~ , U] = size(Lxu);
b=zeros(1,U);

% Computing Ht: Ht = H*A
Ht = H*A;
%---------------------------------------------------------------------------
---



% Computing Rf, Rbinv, Gbar
Rf = Ht' * Ht;
Rbinv = Rf + eye(size(Rf));
Gbar = chol(Rbinv);

% Computing the matrices of interest
G = inv(diag(diag(Gbar)))*Gbar;
S0 = diag(diag(Gbar))*diag(diag(Gbar));
W = inv(S0)*inv(G');

GU = eye(size(G)) + S0*pinv(S0-eye(size(G)))*(G-eye(size(G)));
WU = pinv(S0-eye(size(G)))*inv(G');
MSWMFU = WU*Ht';
index=0;
for u=1:U
  for l=1:Lxu(u)
  b(u) = b(u)+ (1/cb)*log2(S0(index+l,index+l));
  end
  index=index+Lxu(u);
end
```

channel Rxx1/2 1 cplx 2 real

#/user xmit antennas

H=[5 2 ; 3 1];
[b, GU, WU, S0, MSWMFU] = mu_mac(H, **eye(2)**, [1 1] , 2);

b = 2.5646 0.1141

GU =

1.0000 0.3824
0 1.0000

WU =
0.0294 0
-2.1667 5.8333

S0 =
35.0000 0
0 1.1714

MSWMFU =
0.1471 0.0882
0.8333 -0.6667

>> MSWMFU*H =
1.0000 0.3824
2.1667 1.0000

>> SNR = 10*log10(diag(S0)) =
15.4407
0.6872

>> sum(b) = 2.6788

Example 2: 2 x 3 MAC (secondary users)

```
H=[5 2 1
3 1 1];   basically added a 3rd user
[b, GU, WU, S0, MSWMFU] = mu_mac(H, eye(3), [1 1 1] , 2)

b =   2.5646      0.1141   0.1137
GU =    1.0000   0.3824    0.2353
             0   1.0000    0.1667
             0        0    1.0000
WU =
   0.0294       0       0
  -2.1667   5.8333       0
  -1.2857  -0.1429   5.8571
S0 =
  35.0000       0       0
       0   1.1714       0
       0        0   1.1707
MSWMFU =
   0.1471   0.0882
   0.8333  -0.6667
  -0.8571   1.8571
>> sum(b) =   2.7925
>> MSWMFU*H=
   1.0000   0.3824   0.2353
   2.1667   1.0000   0.1667
   1.2857   0.1429   1.0000
>>  SNR10*log10(diag(S0))=
  15.4407
   0.6872
   0.6846
```

- The channel rank is 2 so at least 1 secondary comp = 3-2
- But secondary applies to energy-sum MAC (which this is not, yet)
- Original 2 units of energy spread over 3 users?

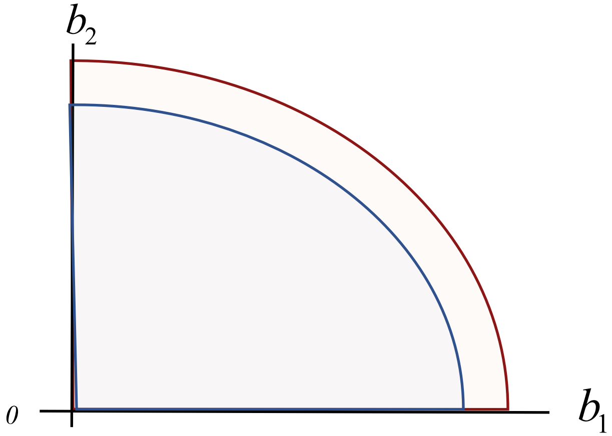
```
>> [b, GU, WU, S0, MSWMFU] = mu_mac(H, (2/3)*eye(3), [1 1 1] , 2)

b =   2.0050   0.1009   0.0696
GU =
   1.0000   0.3824   0.2353
        0   1.0000   0.3878
        0        0   1.0000
WU =
   0.0662        0         0
  -2.3878   6.6582         0
  -2.0000  -0.5000    9.8750
S0 =
  16.1111        0         0
        0   1.1502         0
        0        0    1.1013
MSWMFU =
   0.2206   0.1324
   0.9184  -0.3367
  -0.7500   2.2500
>> sum(b) =  2.1755  (lower than 2x2 value of 2.6788)
```

- Relatively more energy on secondary-user comp(s), bsum ↓

Non-Zero Gap Achievable Region

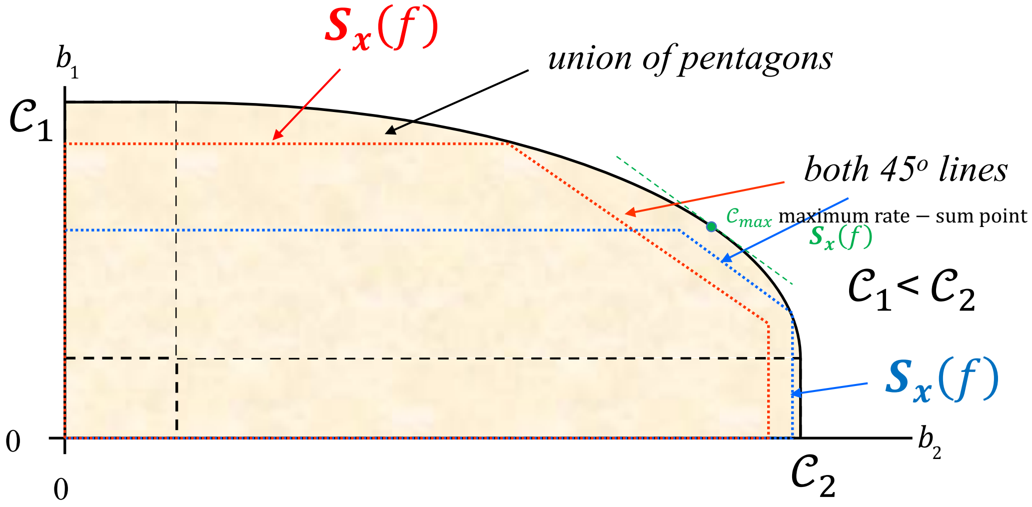
- Construct $\mathcal{C}(\boldsymbol{b})$ with $\Gamma = 0$ dB
- Reduce all rates by γ_b relative to boundary points
- Inscribes smaller region $\mathcal{C}(\boldsymbol{b})$- $(\gamma_b \odot \mathbf{1})$
- Square constellations instead of spheres (AWGN) loss 1.53 dB in gap above (0.25 bit/dimension)



Capacity Region for frequency-indexed channels

Sections 2.7.4.1-2

$\mathcal{C}(b)$ is Union of $S_x(f)$-indexed Pentagons



$$\bar{b} = \sum_{u=1}^{U} \bar{b}_u \leq \bar{\mathcal{I}}(\boldsymbol{x};\boldsymbol{y}) = \int_{-\infty}^{\infty} \frac{1}{2} \cdot \log_2 \left[1 + \frac{\sum_{u=1}^{U} S_{x,u}(f) \cdot |H_u(f)|^2}{S_n(f)} \right] df$$

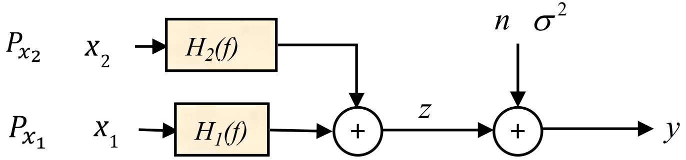
$$S_{x,u}(f) + \frac{\sigma^2 + \sum_{i \neq u} S_{x,i}(f) + |H_i(f)|^2}{|H_u(f)|^2} = K_u$$

Simultaneous water-filling
→ Maximum rate sum

- Each pentagon corresponds to an $S_x(f)$ choice.
  - The pentagons become triangles for the sum-energy MAC.

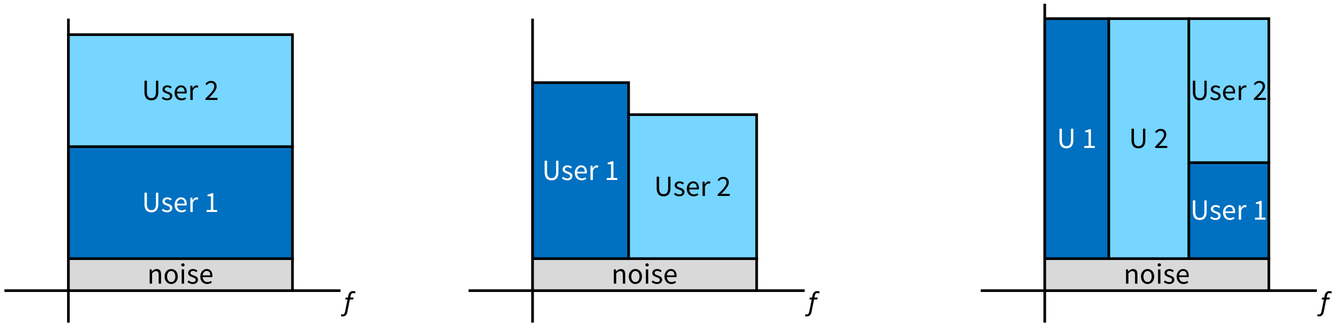
- The union (convex hull is union when inputs are Gaussian) can dimension-share in frequency as $N \to \infty$.

MT MAC

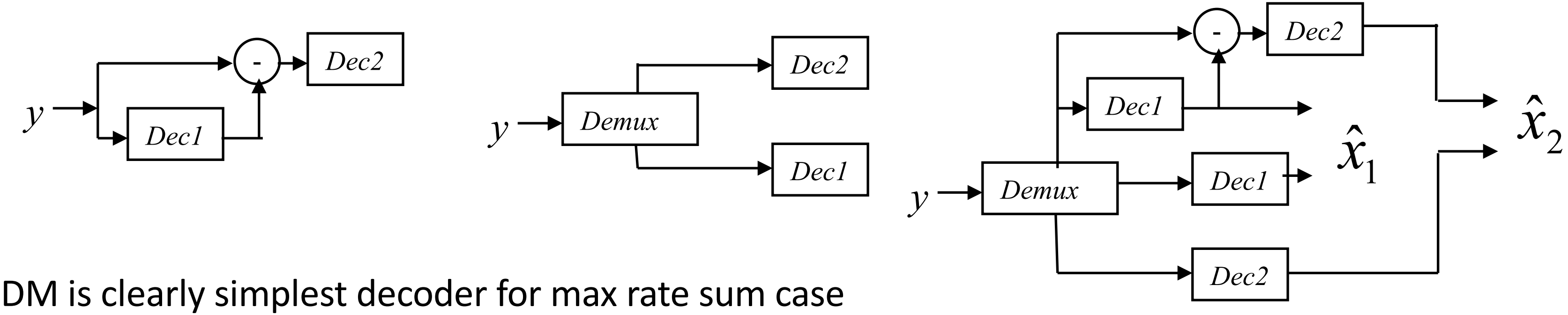


- The users have continuous-time/frequency channels → use MT on each, theoretically
- This really means dimensionality is infinite (or very large) so “dimension-sharing” may be inherent
- SWF applies, but with some interpretation (like power instead of energy, etc and power per dimension instead of power-spectral density, etc.)

Decoders and SWF

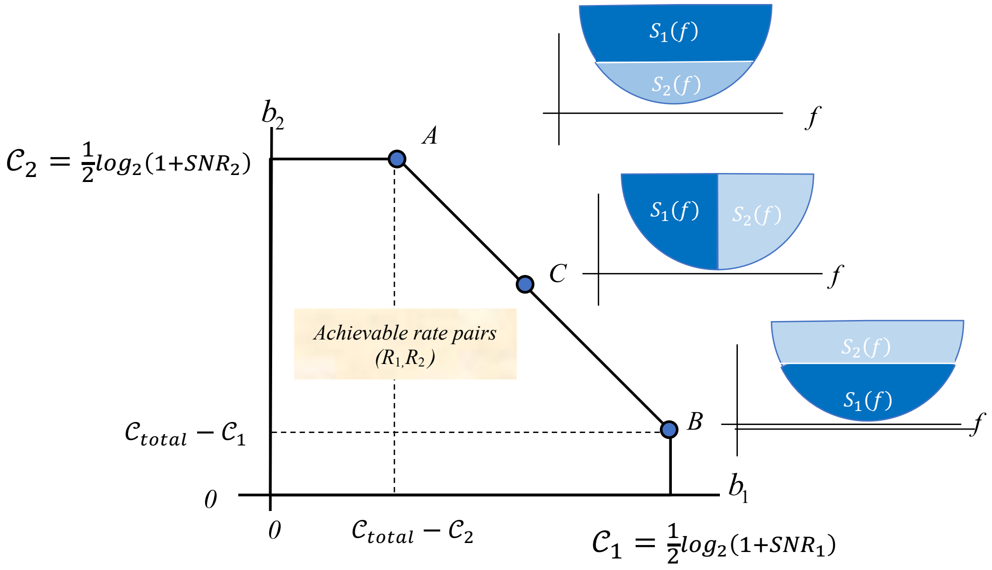


a). both flat *b). FDM* *c). mixed*



- FDM is clearly simplest decoder for max rate sum case

Symmetric 2-user channel and SWF



- Symmetric means $H_1(f) = H_2(f)$ (noise is one-dimensional and added to sum)
- Each of points A, B, and C have different SWF spectra – all have same (max) rate sum

End Lecture 8