STANFORD

Lecture 10

BC, Interference & General MU Channels

May 4, 2026

JOHN M. CIOFFI

Hitachi Professor Emeritus of Engineering

Instructor EE379B – Spring 2026

Announcements & Agenda

- Announcements
  - PS #5 due May 14

- Agenda
  - Maximum BC rate sum
  - Scalar Duality (BC and MAC)
  - Continuous-time scalar BC
  - MAC-set approach to IC
    - Examples
  - IC maximum rate sum and energy sums
  - Nesting, DAS, cellfree, & relay

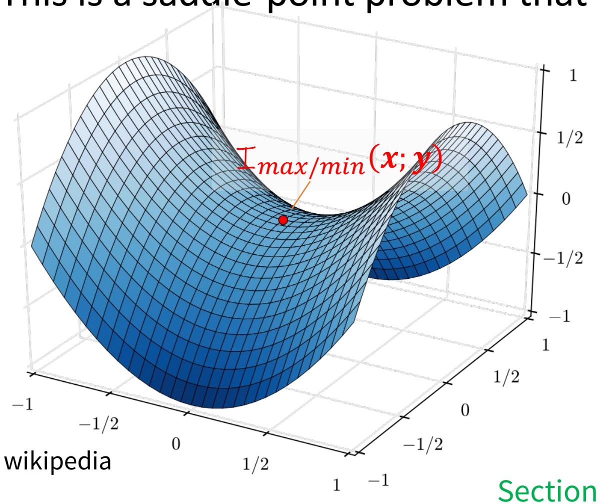
Maximum BC rate sum

Maximum BC rate sum

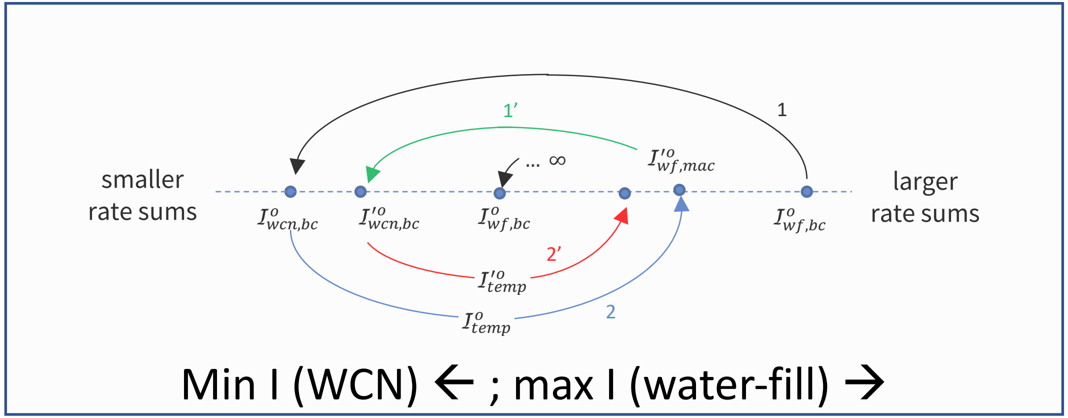
- **Maximize** $\mathcal{I}(\boldsymbol{x};\boldsymbol{y})$ through water-filling (but … presumes receivers can coordinate).
  - This is concave problem that always can be solved for the best input autocorrelation $R_{\boldsymbol{xx}}$.

- **Minimize** $\mathcal{I}_{min}(\boldsymbol{x};\boldsymbol{y})$ through worst-case-noise to get $\mathcal{I}_{wcn}(\boldsymbol{x};\boldsymbol{y})$.
  - This is a convex problem that always can be solved for worst noise autocorrelation R_{wcn}.

- This is a saddle-point problem that produces a max-min = min-max:



wikipedia



bcmax.m

```
function [Rxx, Rwcn, bmax] = bcmax(iRxx, H, Lyu)

 Uses cvx_wcnoise.m and rate-adaptive waterfill.m (Lagrange
 Multiplier based)
  Inputs:
    - iRxx: initial input autocorrelation array, size is Lx x Lx x N.
       Only the sum of traces matters, so can initialize to any valid
       autocorrelation matrix Rxx to run wcnoise.
       needs to include factor N/(N+nu) if nu ~= 0
    - H: channel response, size is Ly x Lx x N, w/o sqrt(N)
    normalization
    - Lyu: array number of antennas at each user; scalar Lyu means same for all

  Outputs:
    - Rxx: optimized input autocorrelation, Lx x Lx x N
    - Rwcn: optimized worst-case noise autocorrelation, with white local noise
          Ly x Ly x N
          so IF H is noise-whitened for Rnn, then actual noise is
          Rwcn^(1/2)*Rnn*Rwcn^(*/2)
    - b: maximum sum rate/real-dimension - user must mult by 2 for
        complex case
```

- Revisit example from slide L9:24

```
H = [    80    70
         50    60 ];
>> iRxx=[1 .8
      .8 1];
>> [Rxx, Rwcn, bmax] = bcmax(iRxx, H, Lyu)

Rxx =
   1.0001   0.0082
   0.0082   0.9999
Rwcn =
   1.0000   0.0049
   0.0049   1.0000
bmax =   10.3517 > 9.6430
```

- Usually converges pretty quickly, not always though – CVX can get finicky when singularity involved.

Revisit example from L9:26

```
iRxx =
   3    0    0
   0    4    0
   0    0    2
>> H =
  80   60   40
  60   45   30
  20   20   20
[RxxA, RwcnA, bmax] = bcmax(iRxx, H, 1)
RxxA =
  3.7515   1.5032  -0.7451
  1.5032   1.5019   1.5007
 -0.7451   1.5007   3.7465
RwcnA =
  1.0000   0.7500   0.0008
  0.7500   1.0000   0.0006
  0.0008   0.0006   1.0000
bmax =  12.1084  (> 11.3777 that occurred earlier)
```

- Secondary components' energy is zeroed for this maximum rate sum.

New Example – Singular Rwcnopt

```
>> H
   1.0719  -0.8627  -0.1901   0.2952
   1.0498  -0.7245   0.2568   0.2757
  -0.4586   0.5595   1.0027   0.0530
   0.4107   0.0496   0.3965  -0.7740
[F,L,M]=svd(H);
L =
   2.0671        0        0        0
        0   1.1449        0        0
        0        0   0.8130        0
        0        0        0   0.0000
H=F(:,1)*M(:,1)'*L(1,1)+ F(:,2)*M(:,2)'*L(2,2)+F(:,3)*M(:,3)'*L(3,3);

[Rxxopt, Rwcnopt, bmax] = bcmax(eye(4), H, 1)

Rxxopt =
   1.1559  -0.7381   0.0995  -0.0691
  -0.7381   0.6399   0.2862  -0.2207
   0.0995   0.2862   1.3091  -0.0326
  -0.0691  -0.2207  -0.0326   0.8951
Rwcnopt =
   1.0000   0.9163  -0.4792  -0.0191
   0.9163   1.0000  -0.0991   0.1251
  -0.4792  -0.0991   1.0000   0.1044
  -0.0191   0.1251   0.1044   1.0000
bmax =   2.1911
>> det(Rwcnopt) % =   1.5337e-09
```

Singular Rwcn

```
inv(Rwcnopt) - inv(H*Rxxopt*H'+Rwcnopt)
-69.9931  62.4186 -26.7687  -6.3512
 62.4186 -54.9656  23.8385   5.6560
-26.7687  23.8385  -9.5873  -2.4256
 -6.3512   5.6560  -2.4256  -0.1050
[Rwcnopt, sumRatebar, S1, S2, S3, S4] =
cvx_wcnoise(Rxxopt, H, [1 1 1 1])
Rwcnopt =
   1.0000   0.9163  -0.4792  -0.0191
   0.9163   1.0000  -0.0991   0.1251
  -0.4792  -0.0991   1.0000   0.1044
  -0.0191   0.1251   0.1044   1.0000
sumRatebar =   2.1911
rank(H) =   3
>> S3+S4(1:4,1:4) =
   0.0978        0        0        0
        0   0.6204        0        0
        0        0   0.6360        0
        0        0        0   0.4705
rank(H) =   3
>> Htilde=pinv(Rwcnopt)*H;
>> [R,Q,P]=rq(Htilde);
R =
   0.0000   0.0211  -0.2595   0.1213
        0  -0.9411  -0.0435   0.0071
        0        0  -1.0542   0.0496
        0        0        0  -1.1499
P =   1   4   2   3
```

```
>> [V,D]=eig(Rxxopt)

V =

  -0.5342  -0.7457  -0.3635  -0.1626
  -0.7873   0.6052  -0.0344  -0.1124
   0.2071   0.2556  -0.9213   0.2073
  -0.2278  -0.1108   0.1336   0.9581

D =

   0.0000        0        0        0
        0   1.7105        0        0
        0        0   1.3639        0
        0        0        0   0.9256
```

Confirms that best input is also singular – it should never have higher rank than number of primary user (components).

Scalar Duality
(BC and MAC)

PS5.2 - 2.29 scalar BC region

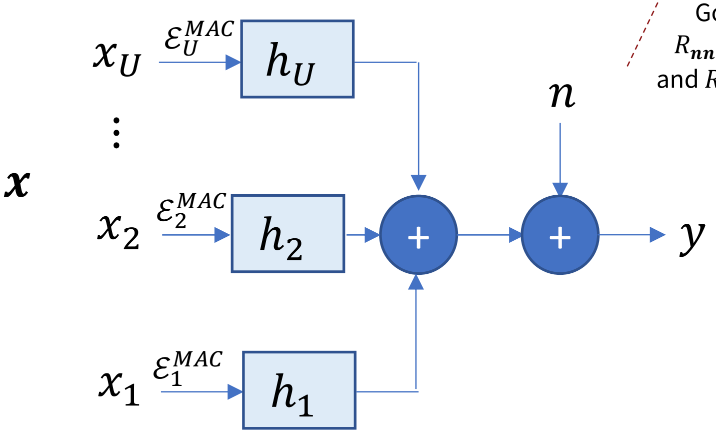
Scalar Dual Channels – Same $I(x;y)$

- Dual Channels have:
  - H_{MAC} for MAC
  - H_{BC} = $H^*_{MAC} \cdot J$ for dual-of-MAC as a BC.

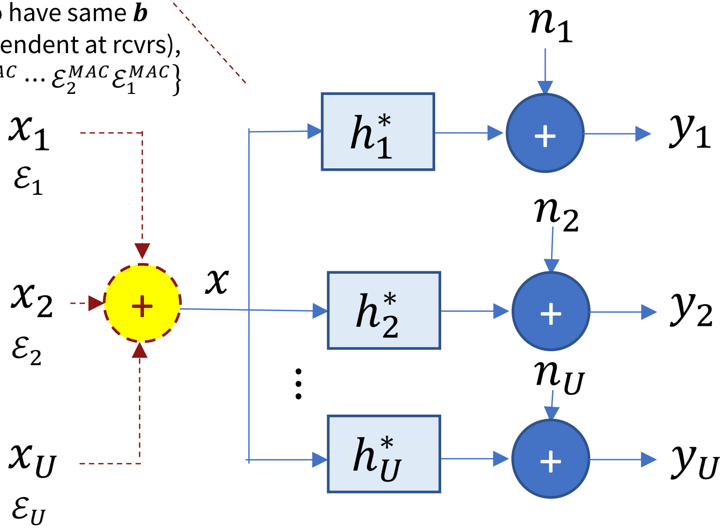
$$\mathcal{I}(x;y) = \mathcal{I}_{MAC} = \log_2\left|H_{MAC} \cdot diag\{\mathcal{E}_U^{MAC} \cdots \mathcal{E}_2^{MAC}\mathcal{E}_1^{MAC}\} \cdot H^*_{MAC} + I\right|$$

$$\mathcal{I}_{BC}(x;y) = \sum_{u=1}^{U} \log_2\left(\frac{\mathcal{E}_u^{BC}\cdot|h_u|^2}{1+|h_u|^2\cdot\sum_{i=u+1}^{U}\mathcal{E}_i^{BC}} + 1\right)$$

$$H_{MAC} = [h_U \; \cdots h_2 \; h_1]$$



Goal: For duals to have same $\boldsymbol{b}$
R_{nn} is white (independent at rcvrs),
and $R_{xx} = diag\{\mathcal{E}_U^{MAC} \cdots \mathcal{E}_2^{MAC}\mathcal{E}_1^{MAC}\}$



$$H_{BC} = \begin{bmatrix} h_1^* \\ h_2^* \\ \vdots \\ h_U^* \end{bmatrix}$$

$$\begin{bmatrix} \mathcal{E}_U^{MAC} \\ \vdots \\ \mathcal{E}_1^{MAC} \end{bmatrix} \preccurlyeq \boldsymbol{\mathcal{E}} \quad \sum_{u=1}^{U} \mathcal{E}_u^{BC} = \mathcal{E}_x = \sum_{u=1}^{U} \mathcal{E}_u^{MAC}$$ (scalar case)

Scalar Duality

- Set data rates equal and solve for $\mathcal{E}_u^{MAC/BC}$:

order intentionally reversed

MAC	BC
$\bar{b}_1 = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_1^{MAC}\cdot g_1}{1+\mathcal{E}_2^{MAC}\cdot g_2+\ldots+\mathcal{E}_U^{MAC}\cdot g_U}\right)$	$\bar{b}_1 = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_1^{BC}\cdot g_1}{1}\right)$
$\bar{b}_2 = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_2^{MAC}\cdot g_2}{1+\mathcal{E}_3^{MAC}\cdot g_3+\ldots+\mathcal{E}_U^{MAC}\cdot g_U}\right)$	$\bar{b}_2 = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_2^{BC}\cdot g_2}{1+\mathcal{E}_1^{BC}\cdot g_2}\right)$
$\vdots$	$\vdots$
$\bar{b}_U = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_U^{MAC}\cdot g_U}{1}\right)$	$\bar{b}_U = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_U^{BC}\cdot g_U}{1+\left[\mathcal{E}_1^{BC}+\ldots+\mathcal{E}_{U-1}^{BC}\right]\cdot g_U}\right)$

Corresponding Energies

$$
\begin{aligned}
\mathcal{E}_1^{BC} &= \mathcal{E}_1^{MAC} \cdot \frac{1}{1 + \mathcal{E}_2^{MAC} \cdot g_2 + \ldots + \mathcal{E}_U^{MAC} \cdot g_U} \\
\mathcal{E}_2^{BC} &= \mathcal{E}_2^{MAC} \cdot \frac{1 + \mathcal{E}_1^{BC} \cdot g_2}{1 + \mathcal{E}_3^{MAC} \cdot g_3 + \ldots + \mathcal{E}_U^{MAC} \cdot g_U} \\
\vdots &= \vdots \\
\mathcal{E}_U^{BC} &= \mathcal{E}_U^{MAC} \cdot \left(1 + \left[\mathcal{E}_1^{BC} + \ldots + \mathcal{E}_{U-1}^{BC}\right] \cdot g_U\right)
\end{aligned}
$$

- By selecting these energies, all user rates are the same (with the order reversal) and running through all such energies that sum to total produces the SAME energy-sum capacity region.

- See proof in text (Theorem 2.8.2 in Section 2.8.4).

Revisit Scalar Example

- Total energy is 1, instead use dual MAC to investigate BC with:
  - $\mathcal{E}_2^{BC} = 0.25$ (bottom of BC),
  - $\mathcal{E}_1^{BC} = 0.75$ (top BC), &
  - reversing order $g_1 = 6400$ and $g_2 = 2500$.

$$\mathcal{E}_2^{MAC} = \frac{\mathcal{E}_2^{BC}}{1+\mathcal{E}_1^{BC}\cdot g_2} = \frac{.25}{1+2500\cdot(.75)} = \frac{1}{7504} = 1.3326 \times 10^{-4} \text{ (top MAC)}$$

$$\mathcal{E}_1^{MAC} = \mathcal{E}_1^{BC} \cdot \left(1 + g_2 \cdot \mathcal{E}_2^{MAC}\right) = .75 \cdot (1 + 2500/7504) = \frac{7503}{7504} = .9999 = 1 - \mathcal{E}_2^{MAC} \text{ (bottom MAC)}$$

- User data rates for this combination are (and were in earlier table found directly for BC).

$$b_1 = \frac{1}{2} \cdot \log_2\left(1 + \frac{\mathcal{E}_1^{MAC} \cdot g_1}{1 + \mathcal{E}_2^{MAC} \cdot g_2}\right) = 6.1144$$

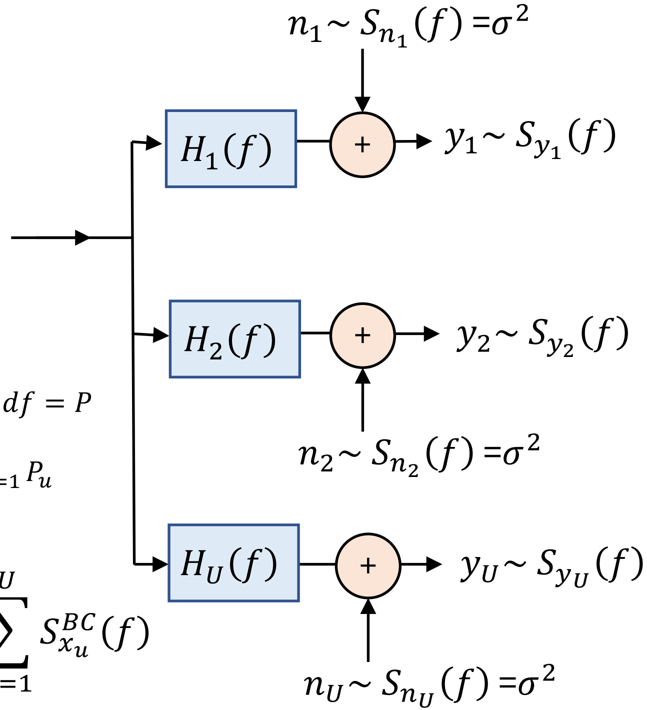
$$b_2 = \frac{1}{2} \cdot \log_2\left(1 + \frac{\mathcal{E}_2^{MAC} \cdot g_2}{1}\right) = .2074$$

- Can use the easier MAC developments to analyze the BC through duality.

Continuous-time Scalar BC

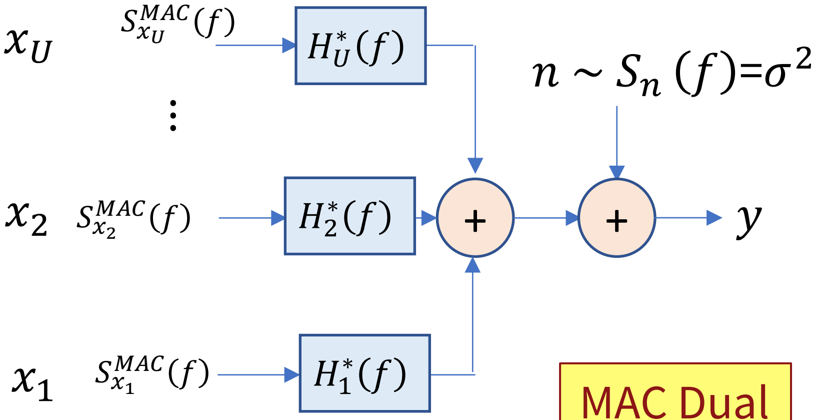
Section 2.8.5

Continuous time/freq Scalar BC



$$\int_{-\infty}^{\infty} S_x(f) \cdot df = P = \sum_{u=1}^{U} P_u$$

$$S_x(f) = \sum_{u=1}^{U} S_{x_u}^{BC}(f)$$



MAC Dual

$\boldsymbol{x}$

$$\sum_{u=1}^{U} \int_{-\infty}^{\infty} S_{x_u}^{MAC}(f) \cdot df = P$$

$$\ln S_{0,u} = \int_{-\infty}^{\infty} \ln\left[1 + \frac{|H_u(f)|^2}{S_{n_u}(f)}\right] \cdot df \;\; ; \;\; (if < \infty)$$

$$SNR_{geo,u} = P_u \cdot S_{0,u}$$

Design for this MAC, and then find dual

Scalar Duality

- Replace with integrals and $\mathcal{E}_u^{MAC/BC} \to S_u^{MAC/Bc}(f)$

order intentionally reversed

MAC	BC
$b_1 = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_1^{MAC}\cdot g_1}{1+\mathcal{E}_2^{MAC}\cdot g_2+\ldots+\mathcal{E}_U^{MAC}\cdot g_U}\right)$	$b_1 = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_1^{BC}\cdot g_1}{1}\right)$ 1 ND user
$b_2 = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_2^{MAC}\cdot g_2}{1+\mathcal{E}_3^{MAC}\cdot g_3+\ldots+\mathcal{E}_U^{MAC}\cdot g_U}\right)$	$b_2 = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_2^{BC}\cdot g_2}{1+\mathcal{E}_1^{BC}\cdot g_2}\right)$
$\vdots$	$\vdots$
$b_U = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_U^{MAC}\cdot g_U}{1}\right)$ 1 ND user	$b_U = \frac{1}{2}\log_2\left(1 + \frac{\mathcal{E}_U^{BC}\cdot g_U}{1+\left[\mathcal{E}_1^{BC}+\ldots+\mathcal{E}_{U-1}^{BC}\right]\cdot g_U}\right)$

Corresponding PSD's

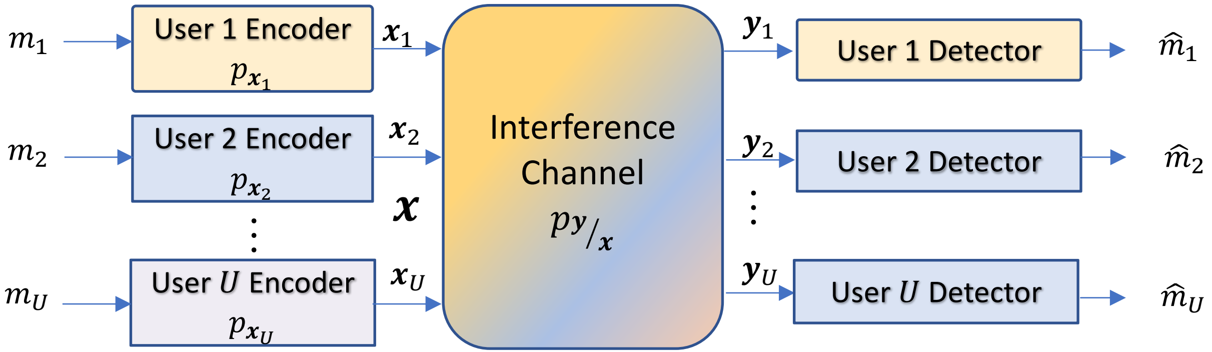
- $\mathcal{E}_u^{MAC/BC} \rightarrow S_u^{MAC/BC}(f)$
- See proof in notes (Theorem 2.8.2 in Section 2.8.4), but execute with PSD's $S_u^{MAC/BC}(f)$

$$
\begin{aligned}
\mathcal{E}_1^{BC} &= \mathcal{E}_1^{MAC} \cdot \frac{1}{1+\mathcal{E}_2^{MAC}\cdot g_2 + \ldots + \mathcal{E}_U^{MAC}\cdot g_U} \\
\mathcal{E}_2^{BC} &= \mathcal{E}_2^{MAC} \cdot \frac{1+\mathcal{E}_2^{BC}\cdot g_2}{1+\mathcal{E}_3^{MAC}\cdot g_3 + \ldots + \mathcal{E}_U^{MAC}\cdot g_U} \\
\vdots &= \vdots \\
\mathcal{E}_U^{BC} &= \mathcal{E}_U^{MAC} \cdot \left(1 + \left[\mathcal{E}_1^{BC} + \ldots + \mathcal{E}_{U-1}^{BC}\right]\cdot g_U\right)
\end{aligned}
$$

MAC-set Approach to IC

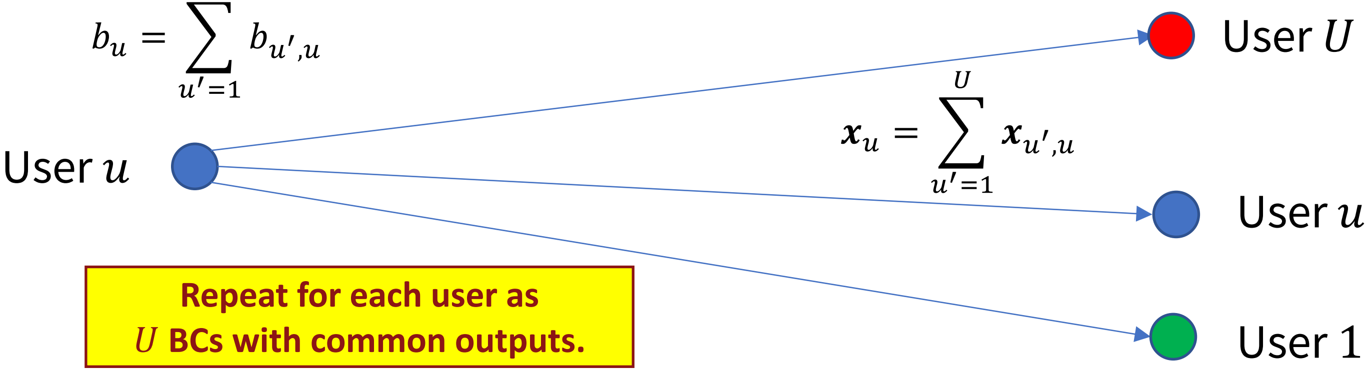
Sec 2.9

The Interference Channel (IC)



Refine Capacity Region Def: with order-approach: Assume each user must use a single-user $\Gamma = 0\ dB$ Code.

$$b_u = \sum_{u'=1}^{U} b_{u',u}$$



$$\boldsymbol{x}_u = \sum_{u'=1}^{U} \boldsymbol{x}_{u',u}$$

U^2 potential sub-user channels/ decodes

Repeat for each user as U BCs with common outputs.

Also could be U MACs with common inputs. u' is receiver index

Prior-User Set (repeat from L7)

- Order vector and its inverse are: $\boldsymbol{\pi}_u = \begin{bmatrix} \pi(U') \\ \vdots \\ \pi(1) \end{bmatrix}$ $\boldsymbol{\pi}_u^{-1} = \begin{bmatrix} U' \\ \vdots \\ 1 \end{bmatrix}$ $j = \pi(i) \rightarrow i = \pi^{-1}(j)$
- Prior-User Set is $\mathbb{P}_u(\boldsymbol{\pi}) = \{j \mid \boldsymbol{\pi}^{-1}(j) < \boldsymbol{\pi}^{-1}(u)\}$
  - That is "all the users before the desired user u in the given order $\boldsymbol{\pi}$.
  - Receiver u best decodes these "prior" users and removes them, while "post" users are noise
  - $\boldsymbol{\pi}$ can be any order in $\mathbb{P}_u(\boldsymbol{\pi})$, but the most interesting is usually $\boldsymbol{\pi}_u$ (receiver u's order)

rcvr/ User i	$\boldsymbol{\pi}_4(i)$	$\boldsymbol{\pi}_3(i)$	$\boldsymbol{\pi}_2(i)$	$\boldsymbol{\pi}_1(i)$
$i = 4$	3	3	4	3
$i = 3$	4	2	3	2
$i = 2$	1	4	2	1
$i = 1$	2	1	1	4
$\mathbb{P}_u(\boldsymbol{\pi}_u)$	{1,2}	{2,4,1}	{1}	{4}

$$\boldsymbol{\Pi} = \begin{bmatrix} 3 & 3 & 4 & 3 \\ 4 & 2 & 3 & 2 \\ 1 & 4 & 2 & 1 \\ 2 & 1 & 1 & 4 \end{bmatrix}$$

really should be 16 with full subusers, but simplify here to 4

- Data rates (mutual information bounds) average only those users who are not cancelled are xtalk noise.

$\mathfrak{I}$	$\mathfrak{I}_4$	$\mathfrak{I}_3$	$\mathfrak{I}_2$	$\mathfrak{I}_1$
top	∞	$\mathcal{I}_3(3/1,2,4)$ 20	∞	∞
	$\mathcal{I}_4(4/1,2)$ 10	$\mathcal{I}_3(2/1,4)$ 9	∞	∞
	$\mathcal{I}_4(1/2)$ 5	$\mathcal{I}_3(4/1)$ 4	$\mathcal{I}_2(2/1)$ 4	$\mathcal{I}_1(1/4)$ 2
bottom	$\mathcal{I}_4(2)$ 1	$\mathcal{I}(1)$ 2	$\mathcal{I}_2(1)$ 2	$\mathcal{I}_1(4)$ 5

$$\mathcal{I}_{min}(\boldsymbol{\Pi}, p_{\boldsymbol{xy}}) = \begin{bmatrix} 4 \\ 20 \\ 1 \\ 2 \end{bmatrix}$$

Maximum number of subusers U^2

- User u has maximum bit rate, when all other users are given (cancelled): $b_u \leq \mathcal{I}(\boldsymbol{x}_u; \boldsymbol{y}_u / \boldsymbol{x}_{U \backslash u})$
- Although -- this may not be good for the other users $i \neq u$.
  - $\mathcal{I}_{min}(\boldsymbol{\Pi}, p_{\boldsymbol{xy}})$ calculation, given a $\boldsymbol{\Pi}$, precedes a subsequent **convex-hull** search over all $\boldsymbol{\Pi}$ to obtain the achievable region $\mathcal{A}(\boldsymbol{b}, p_{\boldsymbol{xy}})$

convex-hull

$$\boldsymbol{b} = \sum_{i=1}^{U} \alpha_i \cdot \boldsymbol{b}_i \qquad 1 = \sum_{i=1}^{U} \alpha_i$$

No more than U terms needed for U-dimensional $\boldsymbol{b}$

Sum of subcomponents

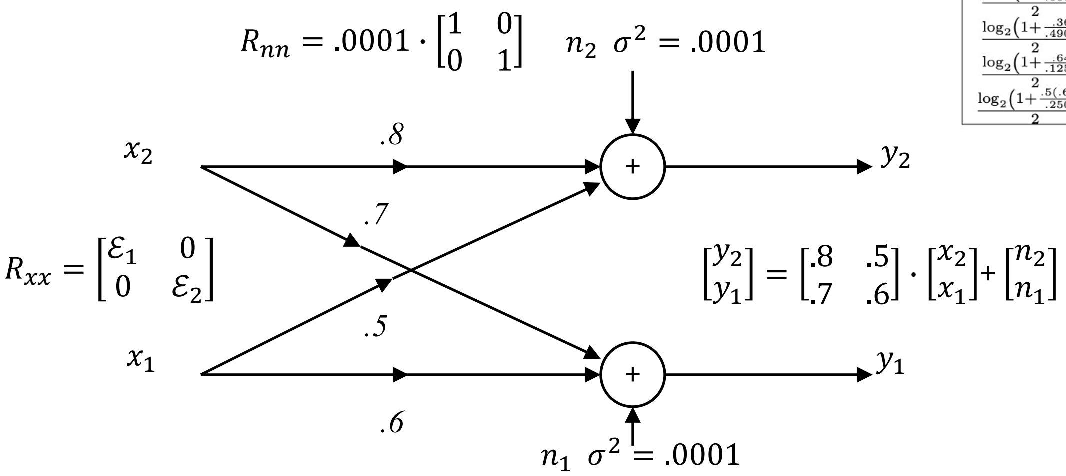
$$\boldsymbol{b} = \begin{bmatrix} b_U \\ \vdots \\ b_1 \end{bmatrix} \qquad b_u = \sum_{i=1}^{U} b_{i,u}$$

Will do this for $\mathcal{I}_{min}(\boldsymbol{\Pi}, p_{\boldsymbol{xy}})$ values over possible $\boldsymbol{\Pi}$'s

- For $U \geq 2$, the other users' subuser components may be desirable to decode, but not all → $U'! \leq (U^2)!$ for **each** receiver's order may need search/evaluation.
- $\boldsymbol{\Pi}$ maximally has $(U'!)^U$ possible choices (in most general case).

At any receiver u , the subuser components separate into two groups for any given order $\boldsymbol{\pi}_u$:
(1) those cancelled (or generally conditional probability has specific given values for those components), and
(2) those not cancelled, which are averaged out generally in marginal distributions.
If that choice is made for each user for each receiver, there are thus maximally U choices across all receivers into (1) or (2), so $U' \leq U^2$.

Example channel – Scalar Gaussian IC



$\frac{\log_2(1+\frac{.64}{.0001})}{2}$ =6.3220	$\frac{\log_2(1+\frac{.36}{.0001})}{2}$ =5.9071	$\frac{\log_2(1+\frac{.64}{.2501})}{2}$ =0.9157	$\frac{\log_2(1+\frac{.25}{.6401})}{2}$ =0.2378
$\frac{\log_2(1+\frac{.36}{.4901})}{2}$ =0.3973	$\frac{\log_2(1+\frac{.49}{.3601})}{2}$ =0.6196	$\frac{\log_2(1+\frac{(.5).64}{.0001})}{2}$ =5.8222	$\frac{\log_2(1+\frac{(.5).36}{.0001})}{2}$ =5.4073
$\frac{\log_2(1+\frac{.64}{.1251})}{2}$ =1.3063	$\frac{\log_2(1+\frac{.18}{.4901})}{2}$ =0.2257	$\frac{\log_2(1+\frac{.49}{.1801})}{2}$ =0.9478	$\frac{\log_2(1+\frac{.36}{.2451})}{2}$ =0.6519
$\frac{\log_2(1+\frac{.5(.64)}{.2501})}{2}$ =0.5944	$\frac{\log_2(1+\frac{.5(.49)}{.3601})}{2}$ =0.3744	$\frac{\log_2(1+\frac{.5(.25)}{.6401})}{2}$ =0.1287	$\frac{\log_2(1+\frac{.25}{.3201})}{2}$ =0.4163

Table 2.6: Some useful calculations for the upcoming example

Convex combos of
These 2 vertices
Probably includes
Other sub user combos
(because they are big)

- Earlier H, but this time as an IC
  - .8>.7 and .6 > .5
  - Not complete set of orders (4 instead of (4!)^2 = 576)
- Shaded points are interior to line formed by unshaded points

$\mathbf{\Pi}$	u	$\mathcal{E}_u$	$\mathbb{P}_u(\boldsymbol{\pi}_u)$	$\mathscr{I}_u(x_u; y_u/\mathbb{P}_u(\boldsymbol{\pi}_u))$	$\mathscr{I}_{\neg u}(x_u; y_{\neg u}/\mathbb{P}_{\neg u}(\boldsymbol{\pi}_{\neg u}))$	$\mathcal{I}_{min,u}(\mathbf{\Pi}, \boldsymbol{\mathcal{E}})$
$\begin{bmatrix}2 & 2\\1 & 1\end{bmatrix}$	2	1	{1}	6.322	∞	6.322
	1	1	$\emptyset$	.3973	.2378	.2378
$\begin{bmatrix}1 & 1\\2 & 2\end{bmatrix}$	2	1	$\emptyset$	.9157	.6196	.6196
	1	1	{2}	5.9071	∞	5.9071
$\begin{bmatrix}1 & 2\\2 & 1\end{bmatrix}$	2	1	$\emptyset$	.9157	∞	.9157
	1	1	$\emptyset$	.3973	∞	.3973
$\begin{bmatrix}2 & 1\\1 & 2\end{bmatrix}$	2	1	{1}	6.322	.2378	.2378
	1	1	{2}	5.9071	.6196	.6196

Table 2.7: Evaluation of $\mathcal{I}_{min}$ for different orders.

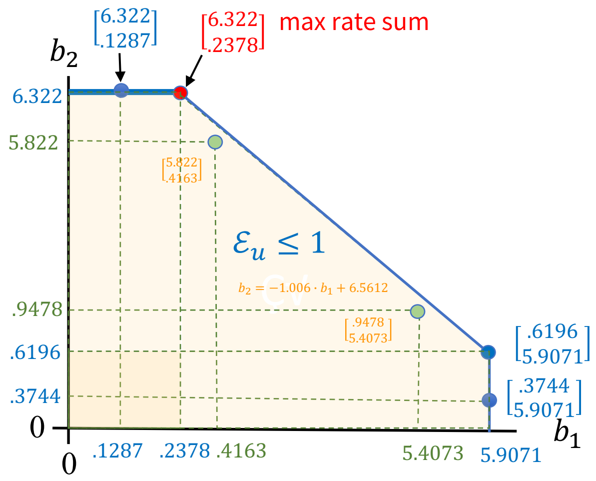
IC Rate Region Examples

Sec 2.9.1

Example continued

$\mathbf{\Pi}$	u	$\mathcal{E}_u$	$\mathbb{P}_u(\boldsymbol{\pi}_u)$	$\mathscr{I}_u(x_u; y_u/\mathbb{P}_u(\boldsymbol{\pi}_u))$	$\mathscr{I}_u(x_{\neg u}; y_{\neg u}/\mathbb{P}_{\neg u}(\boldsymbol{\pi}_{\neg u}))$	$\mathcal{I}_{min,u}(\mathbf{\Pi}, \boldsymbol{\mathcal{E}})$
$\begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix}$	2	1	{1}	6.322	∞	6.322
	1	0.5	$\emptyset$	.2257	.1287	.1287
$\begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}$	2	1	$\emptyset$	1.3063	.9478	.9478
	1	0.5	{2}	5.4073	∞	5.4073
$\begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix}$	2	0.5	{1}	5.822	∞	5.822
	1	1	$\emptyset$	.6519	.4163	.4163
$\begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}$	2	0.5	$\emptyset$	5.822	.3744	.3744
	1	1	{1}	5.9071	∞	5.9071
$\begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix}$	2	1	{1, 2}	6.322	∞	6.322
	1	0.95	{1}	.3818	.2276	.2276
$\begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}$	2	1	{2}	.9425	.6412	.6412
	1	0.95	{1, 2}	5.8701	∞	5.8701

Table 2.8: More example points with the best orders



- The dimension-sharing of the large-b_u points dominates the other points on the interior.
- Also – check of vertices' derivatives relative to the dimension-sharing line (-1.006) – try upper.

$$\ln(2) \cdot \frac{db_2}{d\mathcal{E}_2} = \frac{3200}{6400 \cdot \mathcal{E}_2 + 1} = 0.4999$$

$$\ln(2) \cdot \frac{db_1}{d\mathcal{E}_2} = -\frac{3200 \cdot 2500 \cdot \mathcal{E}_1}{6400 \cdot (\mathcal{E}_2 + 2500 \cdot \mathcal{E}_1 + 1) \cdot (6400 \cdot \mathcal{E}_2 + 1)} = -.1404$$

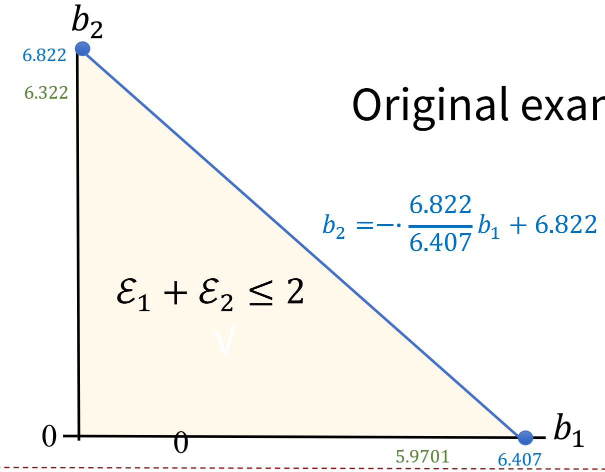
$$\frac{db_2}{db_1} = -3.56$$

Vertices all (○) inside pentagon for this example

- For 2 vertices if magnitude of slope is less than 1, then upper point and otherwise lower point (or whole line).
  - Could check other vertex also, but if curvature is within the line already (convex), then no need.

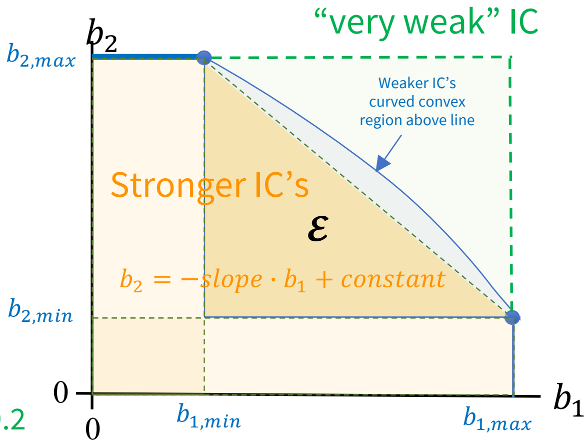
Energy-sum IC extension

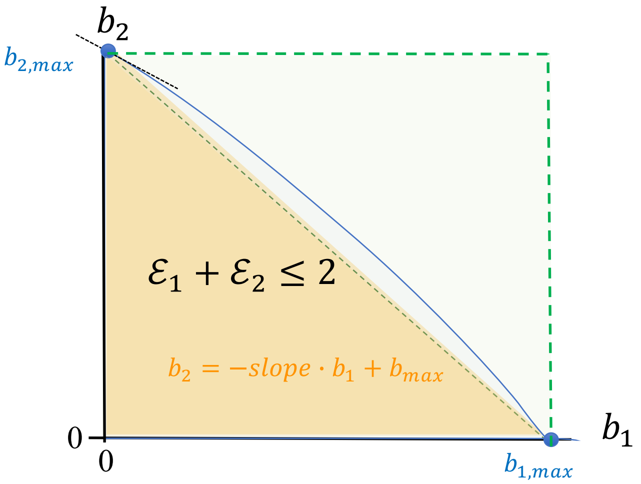
- Same channel with only energy-sum constraint:
- Same derivative test shows dimension-sharing line is boundary



- General derivative test

$$
\begin{aligned}
\ln(2) \cdot \frac{db_2}{d\mathcal{E}_2} &= \frac{g_{22}/2}{g_{22} \cdot \mathcal{E}_2 + 1} \\
\ln(2) \cdot \frac{db_1}{d\mathcal{E}_2} &= -\frac{g_{22}/2 \cdot g_{12} \cdot \mathcal{E}_1}{g_{22} \cdot (\mathcal{E}_2 + g_{12} \cdot \mathcal{E}_1 + 1) \cdot (g_{22} \cdot \mathcal{E}_2 + 1)} \quad \textbf{or} \\
\ln(2) \cdot \frac{db_1}{d\mathcal{E}_2} &= -\frac{g_{21}/2 \cdot g_{11} \cdot \mathcal{E}_1}{g_{21} \cdot (\mathcal{E}_2 + g_{11} \cdot \mathcal{E}_1 + 1) \cdot (g_{21} \cdot \mathcal{E}_2 + 1)} \quad ,
\end{aligned}
$$





Symmetric 2x2 IC

$$\begin{bmatrix} y_2 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 & \alpha \\ \alpha & 1 \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ x_1 \end{bmatrix} + \begin{bmatrix} n_2 \\ n_1 \end{bmatrix}$$

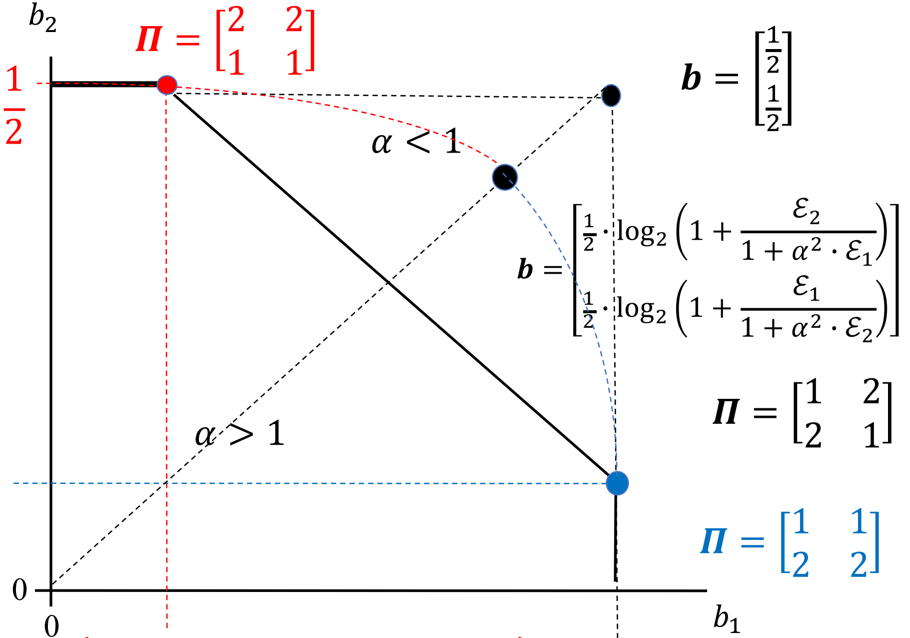
$R_{\boldsymbol{nn}} = I$ and $\boldsymbol{\mathcal{E}} \preccurlyeq \mathbf{1}$

- When $\alpha \to 0$, there is no crosstalk and so $\mathcal{C}_{IC}(\boldsymbol{b})$ is a square.
- When $\alpha > 1$, $\mathcal{C}_{IC}(\boldsymbol{b})$ is a pentagon.
- When $0 < \alpha < 1$, $\mathcal{C}_{IC}(\boldsymbol{b})$ is intermediate

$$\min \left\{ \begin{matrix} \frac{1}{2} \cdot \log_2\left(1 + \frac{\mathcal{E}_2}{1 + \alpha^2 \cdot \mathcal{E}_1}\right) , \\ \frac{1}{2} \cdot \log_2\left(1 + \frac{\alpha^2 \cdot \mathcal{E}_2}{1 + \mathcal{E}_1}\right) \end{matrix} \right\}$$

These two are same for $\alpha = 1$ and equal energy, and determine Imin vector possibilities

Achievable Region when $\mathcal{E}_1 = \mathcal{E}_2 = 1$



$$\min \left\{ \begin{matrix} \frac{1}{2} \cdot \log_2\left(1 + \frac{\mathcal{E}_1}{1 + \alpha^2 \cdot \mathcal{E}_2}\right) , \\ \frac{1}{2} \cdot \log_2\left(1 + \frac{\alpha^2 \cdot \mathcal{E}_1}{1 + \mathcal{E}_2}\right) \end{matrix} \right\}$$

Vector Gaussian IC Example, $L_{x,u} \equiv 1$; $L_{y,u} \equiv 2$

- 2 users and H is **4 x 2:** $\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_2 \\ \boldsymbol{y}_1 \end{bmatrix} = \begin{bmatrix} H_2 \\ H_1 \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ x_1 \end{bmatrix} + \begin{bmatrix} \boldsymbol{n}_2 \\ \boldsymbol{n}_1 \end{bmatrix}$.

$$H_2 = [\boldsymbol{h}_{22} \quad \boldsymbol{h}_{21}] = \begin{bmatrix} .9 & .3 \\ .3 & .8 \end{bmatrix}$$

$R_{\boldsymbol{nn}} = .01 \cdot I$

$$H_1 = [\boldsymbol{h}_{12} \quad \boldsymbol{h}_{11}] = \begin{bmatrix} .8 & .7 \\ .6 & .5 \end{bmatrix}$$

```
>> H2 = [9     3
       3     8 ];
>> Rb2inv=H2'*H2+diag([1 1]);
>> Gbar2=chol(Rb2inv);
>> G2=inv(diag(diag(Gbar2)))*Gbar2;
>> S02=diag(diag(Gbar2))*diag(diag(Gbar2));
>>  0.5*log2(diag(S02)) =
b2 =  3.2539
b1 =  2.7526
>>  H1 = [8     7
        6     5 ];
>> Rb1inv=H1'*H1+diag([1 1]);
>> Gbar1=chol(Rb1inv);
>> S01=diag(diag(Gbar1))*diag(diag(Gbar1));
>> 0.5*log2(diag(S01)) =
b2 =  3.3291
b1 =  0.4128
```

Two MAC sets

Coincides with $\mathcal{I}_{min} = \begin{bmatrix} 3.25 \\ .413 \end{bmatrix}$

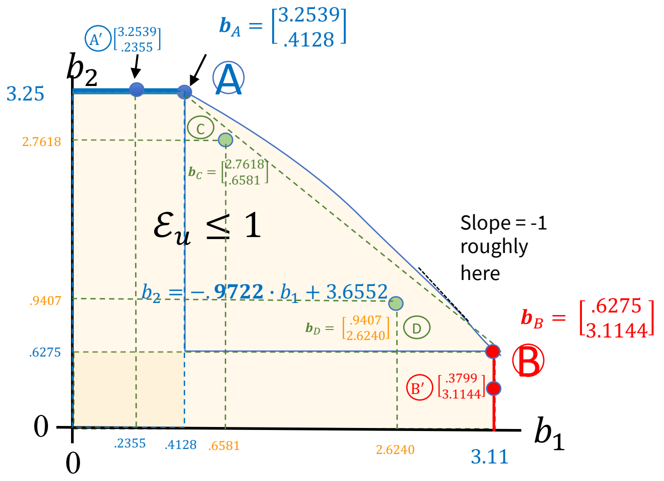
```
>> J2=hankel([0 1]);
>> Rb2inv=J2*H2'*H2*J2+diag([1 1]);
    74    51
    51    91
>> Gbar2=chol(Rb2inv);
>> S02=diag(diag(Gbar2))*diag(diag(Gbar2));
>> 0.5*log2(diag(S02)) =
b1 =  3.1047
b2 =  2.9018

>> Rb1inv=J2*H1'*H1*J2+diag([1 1]);
>> Gbar1=chol(Rb1inv);
>> S01=diag(diag(Gbar1))*diag(diag(Gbar1));
>> 0.5*log2(diag(S01)) =
b1 =  3.1144
b2 =  0.6275
```

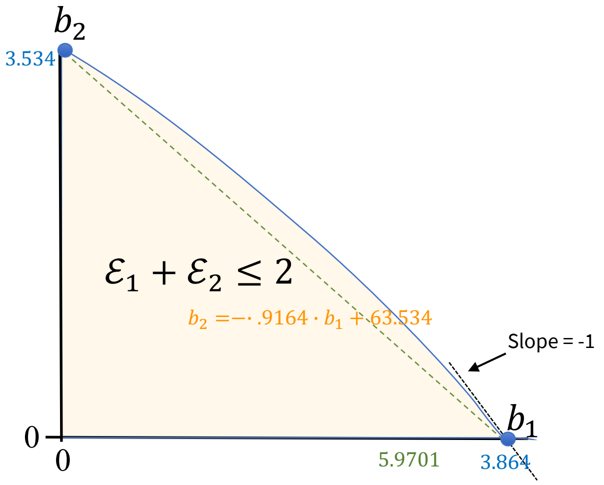
Two MAC Sets with Reversed order

Coincides with $\mathcal{I}_{min} = \begin{bmatrix} 3.11 \\ .628 \end{bmatrix}$

Try various energy points
With the same orders.

4x2 IC Example continued





- Vary energies near max points to see if local points above or below dimension-sharing, so curved boundary or flat
- Since the line has slope magnitude less than 1, then the curvature is above this line with max rate sum at magnitude 1.

PS 5.4 (2.31) – IC channel has mix, one 2x2 user and one scalar user

Chris Adelico-Ferrarin's MU_IC.m

```
function [b, GU, WU, S0, MSWMFU] = mu_ic(H, A, Lxu, Lyu, cb)
 ---------------------------------------------------------------------------
 Per-tonal (temporal dimension) multiuser interference channel
 receiver and per-user bits - Chris Adelico Ferrarin - 2023

 Inputs: H, A, Lxy, Lyu, cb
 Outputs: b, GU, WU, S0, MSWMFU

 Definitions:

 H: noise-whitened channel matrix [HUU ⋯ HU1] sum-Lyu x sum-Lxu
                        |⋮  ⋱  ⋮|
                        |H2U ⋯ H21|
                        [H1U ⋯ H11]
 A: Block Diag sq-root sum-Lxu x sum-Lxu discrete modulators,
   blkdiag([AU ... A1]);  The Au entries derive from each IC user's
   Lxu x Lxu input autocorrelation matrix, where the trace of each such
   autocorrelation matrix is user u's energy/symbol. This is per-tone.

 Lxu: # of input dimensions for each user U ... 1 in 1 x U row vector
 Lyu: # of output dimensions for each user U ... 1 in 1 x U row vector
 cb: = 1 if complex baseband or 2 if real baseband channel

 GU: unbiased feedback matrix sum-Lxu x sum-Lxu x U with matrices indexed
    from user U (e.g. GU(:,:,2) gives GU for user U-1). with
    matrices indexed from user U (e.g. WU(:,:,2) gives WU for user U-1).
 S0: sub-channel channel gains sum-Lxu x sum-Lxu x U with matrices indexed
    from user U (e.g. GU(:,:,2) gives S0 for user U-1).
 MSWMFU: unbiased mean-squared whitened matched filter, sum-Lxu x Ly x U
       with matrices indexed from user U (e.g. MSWMFU(:,:,2) gives

 WU: unbiased feedforward linear equalizer sum-Lxu x sum-Lxu x U MSWMFU for
user U-1).
 b - user u's bits/symbol 1 x U
   the user should recompute b if there is a cyclic prefix
```

Does not Find $\mathcal{I}_{min}$

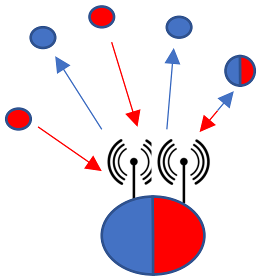
- IC needs Lxu and Lyu
- A is like mu_mac
- Per tone
- Arrange input Hu matrices according to order $\boldsymbol{\Pi}$

```
H2 = [0.9 0.3; 0.3 0.8];    % From L12:6
H1 = [0.8 0.7; 0.6 0.5];
sigma2 = 0.01;
Ht = [H2; H1] / sqrt(sigma2);  % this is 4x2 matrix (2 outputs/input)
A = [1 0; 0 1];
Lxu = [1 1];
Lyu = [2 2];
cb = 2;
>> [b_A, GU_A, WU_A, S0_A, MSWMFU_A] = mu_ic(Ht, A, Lxu, Lyu, cb);
USER 2                          USER 1
>> b_A % =
      3.2539                    0.4128
GU_A(:,:,1) =                   GU_A(:,:,2) =
   1.0000    0.5667              1.0000    0.8600
        0    1.0000                   0    1.0000
WU_A(:,:,1) =                   WU_A(:,:,2) =
   0.0111         0              0.0100         0
  -0.0126    0.0225             -1.1026    1.2949
S0_A(:,:,1) =                   S0_A(:,:,2) =
  91.0000         0            101.0000         0
        0   45.4176                   0    1.7723
MSWMFU_A(:,:,1) =               MSWMFU_A(:,:,2) =
   0.1000    0.0333              0.0800    0.0600
  -0.0460    0.1423              0.2436   -0.1410
```

Order implied by H's column index

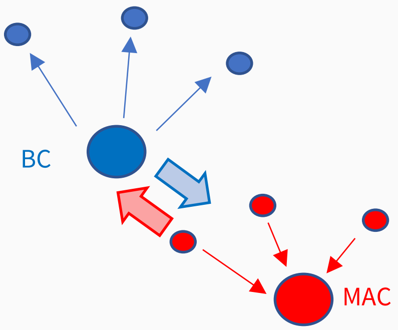
Nesting, DAS, cellfree & Relay

Multiuser Nesting



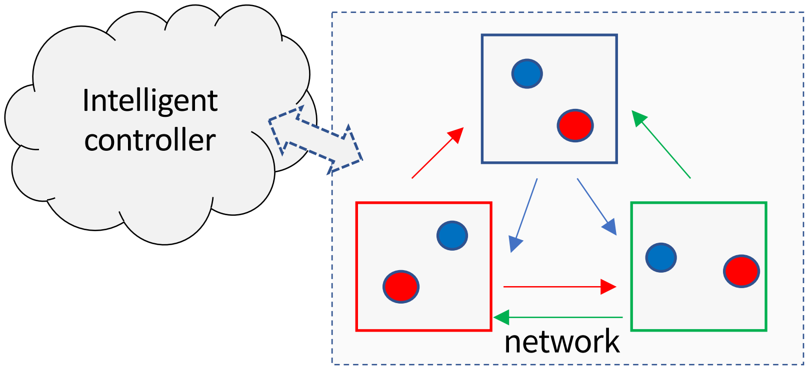
a) radio node edge
(base station or
Access point)

~



b) two subnetworks
Down (BC) and up (MAC)
IC network between them

Macro User



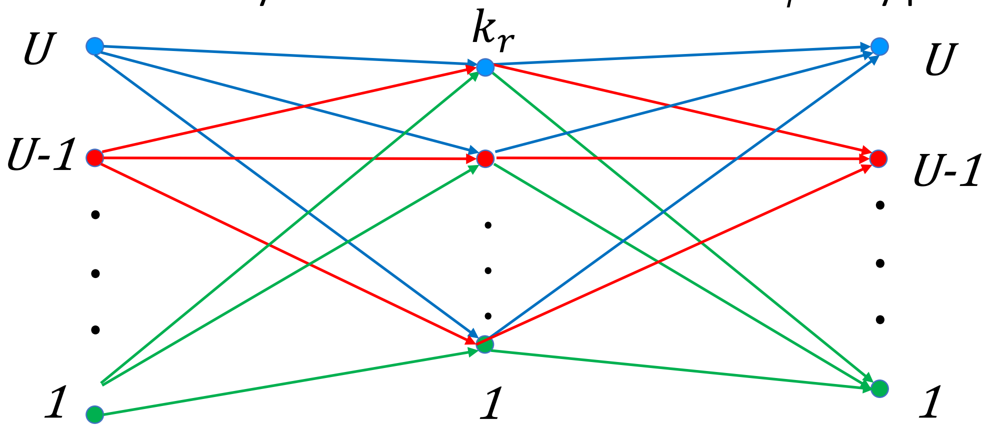
c) three nested IC: (BC, MAC)
Like those in b), nested into
3x3 IC network

**Will be suboptimal,
But easier
Section 5.6.2.3 later**

Single-Stage Relay Channel

- Conceptually uses what we know already and introduces sub-users at k_r relay points

$U \cdot k_r$ subuser components



PS 5.5 (2.32) – simple relay channel

$\boldsymbol{B}^{(1)}$ is $U \times k_r$ MAC user set

$\boldsymbol{B}^{(2)}$ is $k_r \times U$ BC user set

$$\mathcal{A}^{(1)}\left(\boldsymbol{B}^{(1)}, R_{\boldsymbol{xx}}^{(1)}\right) = \cup_{\boldsymbol{\Pi}^{(1)}}^{conv}\left\{\boldsymbol{B}_{min}^{(1)}\left(\boldsymbol{\Pi}^{(1)}, R_{\boldsymbol{xx}}^{(1)}\right)\right\} \qquad \mathcal{A}^{(2)}\left(\boldsymbol{B}^{(2)}, R_{\boldsymbol{xx}}^{(2)}\right) = \cup_{\boldsymbol{\Pi}^{(2)}}^{conv}\left\{\boldsymbol{B}_{min}^{(2)}\left(\boldsymbol{\Pi}^{(2)}, R_{\boldsymbol{xx}}^{(2)}\right)\right\}$$

$\beta_k^{(1)}$ $\quad$ $\beta_u^{(2)}$

$$b_u\left(R_{\boldsymbol{xx}}^{(1)}, R_{\boldsymbol{xx}}^{(2)}\right) = \left\{ b_u \mid b_u \in \sum_{i=1}^{k_r} \min_{\left\{\beta_k^{(1)} \in \boldsymbol{B}_k^{(1)} \wedge \beta_k^{(2)} \in \boldsymbol{B}_k^{(2)}\right\}} \left[\beta_k^{(1)}\left(u, R_{\boldsymbol{xx}}^{(1)}\right) ; \beta_u^{(2)}\left(k, R_{\boldsymbol{xx}}^{(2)}\right)\right]\right\}$$

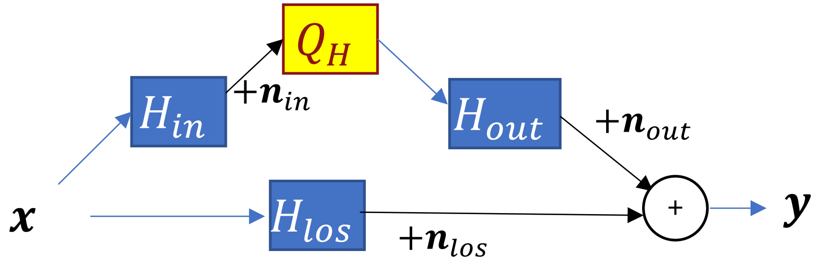
$$\mathcal{C}(\boldsymbol{b}) = \sum_{u'=1}^{U}\left[\cup_{R_{\boldsymbol{xx}}^{(1)} \wedge R_{\boldsymbol{xx}}^{(2)}}^{\boldsymbol{hull}} \ \cup_{\boldsymbol{\Pi}^{(1)} \otimes \boldsymbol{\Pi}^{(2)}}^{\boldsymbol{hull}} \boldsymbol{b}\left(R_{\boldsymbol{xx}}^{(1)}, R_{\boldsymbol{xx}}^{(2)}\right)\right)\right]_{u,u'}$$

- Multi-stage tedious, but same principles apply recursively

Reflective Intelligent Surfaces (RIS)

Posed Project/Research
"maxRIS" or "minRIS"

$$\boldsymbol{y} = \underbrace{\begin{bmatrix} H_{los} \\ H_{out} \cdot Q_H \cdot H_{in} \end{bmatrix}}_{H_{RIS}} \cdot \boldsymbol{x} + \underbrace{\begin{bmatrix} \boldsymbol{n}_{los} \\ \boldsymbol{n}_{out} + Q_H \cdot \boldsymbol{n}_{in} \end{bmatrix}}_{\boldsymbol{n}_{RIS}}$$



- The RIS matrix Q_H satisfies $\|Q_H\|_F^2 \leq G_H$, the RIS gain – it may also satisfy
  - Q_H is unitary matrix (preserves energy)
  - Q_H is diagonal, and usually also unitary, to be phase/gain-only adjustment on each antenna port (in-to-out)
  - Q_H has individual elements restricted
- For a given $R_{\boldsymbol{xx}}$, maximize over Q_H

$$\mathcal{I}(\boldsymbol{y};\boldsymbol{x}) = \log_2 \left| R_{n,RIS} + H_{RIS} \cdot R_{\boldsymbol{xx}} \cdot H_{RIS}^* \right|$$

- For a given Q_H , maximize the same over $R_{\boldsymbol{xx}}$

$$R_{\boldsymbol{nn},RIS} = \begin{bmatrix} R_{\boldsymbol{nn}} & 0 \\ 0 & R_{\boldsymbol{nn},out} + Q_H \cdot R_{\boldsymbol{nn}in} \cdot Q_H^* \end{bmatrix}$$

- Iterate? → not convex in the Q_H , this needs work.
  - However this instructor postulates that $Q_H^{opt} = V_{out}^* \cdot V_{in}^*$
  - where

$$H_{in} \cdot R_{\boldsymbol{xx}} \cdot H_{in}^* + R_{\boldsymbol{n}_{in}\boldsymbol{n}_{in}} = V_{in} \cdot \Lambda_{in} \cdot V_{in}^*$$

$$R_{out}^{-1/2} \cdot H_{out} \cdot \Lambda_{in} \cdot H_{out}^* \cdot R_{out}^{-*/2} + I = V_{out} \cdot \Lambda_{out} \cdot V_{out}^*$$

End Lecture 10