Simulations of Antenna Combining in Multiuser Diversity

This worksheet lets you run simulations of antenna combining algorithms for interference suppression. Both zero forcing and MMSE approaches are supplied. Notation differs a bit from class notes (notably **G,** instead of **C,** for channel gain matrix).

Simulated Adaptive Nulling (Zero Forcing) of Interferers

unit variance complex Gaussian generator:

$$\mathrm{cgauss}(x) := \sqrt{-2\cdot\ln(\mathrm{rnd}(1))}\cdot\exp(j\cdot\mathrm{rnd}(2\cdot\pi))$$

sources (mobiles)

$K := 2$

antennas

$M := 2$

Because the objective is exact nulling, the weight vector for a given user does not depend on any of the user signal strengths. The users are invisible to each other. Of course, each user still experiences an error rate due to noise that depends on its own strength. For convenience, we will give all the users equal amplitude A. Their expected BERs will be equal, so we can average their simulated BERs for more accuracy. The matrix **A** is then proportional to the identity matrix, so **A** = A **I**. The complex gains will all have $\sigma_g^2=1$.

gain matrix generation

$$\mathrm{Ggen}(\mathrm{dummy}) := \left| \begin{array}{l} \text{for } i \in 0..M-1 \\ \quad \text{for } n \in 0..K-1 \\ \quad\quad G_{i,n} \leftarrow \mathrm{cgauss}(i) \\ G \end{array} \right.$$

noise variances after weighting:

$$\mathrm{noisevar}(\Gamma, G) := \left| \begin{array}{l} \mathrm{inv} \leftarrow G^{-1} \\ R_v \leftarrow \Gamma^{-1}\cdot\mathrm{inv}\cdot\overline{\mathrm{inv}^T} \\ \text{for } i \in 0..K-1 \\ \quad \mathrm{var}_i \leftarrow R_{v_{i,i}} \\ \mathrm{Re}(\mathrm{var}) \end{array} \right.$$

useful vector

$k := 0..K-1 \qquad \mathrm{zero}_k := 0$

the simulation

$$\mathrm{BER}(\Gamma, N_{sim}) := \left| \begin{array}{l} \mathrm{sum} \leftarrow \mathrm{zero} \\ \text{for } \mathrm{count} \in 1..N_{sim} \\ \quad \left| \begin{array}{l} G \leftarrow \mathrm{Ggen}(\mathrm{count}) \\ \mathrm{vars} \leftarrow \mathrm{noisevar}(\Gamma, G) \\ P_e \leftarrow \overrightarrow{Q\left(\frac{1}{\sqrt{\mathrm{vars}}}\right)} \\ \mathrm{sum} \leftarrow \mathrm{sum} + P_e \end{array} \right. \\ \dfrac{\sum \mathrm{sum}}{N_{sim}\cdot K} \end{array} \right.$$

$$\mathrm{pole\_ratio}(\Gamma_r) := \frac{\sqrt{1+\Gamma_r^{-1}}+1}{\sqrt{1+\Gamma_r^{-1}}-1} \qquad \mathrm{BC}(n,k) := \frac{n!}{n-k!\cdot k!}$$

binomial coefficient

$$P_{noCCI}(\Gamma_r, M) := \frac{1}{\left(1+\mathrm{pole\_ratio}(\Gamma_r)\right)^{2\cdot M-1}} \cdot \sum_{k=0}^{M-1} \mathrm{BC}(2\cdot M-1, k)\cdot \mathrm{pole\_ratio}(\Gamma_r)^k$$

$\mathrm{BER}(20, 1000) = 0.012$

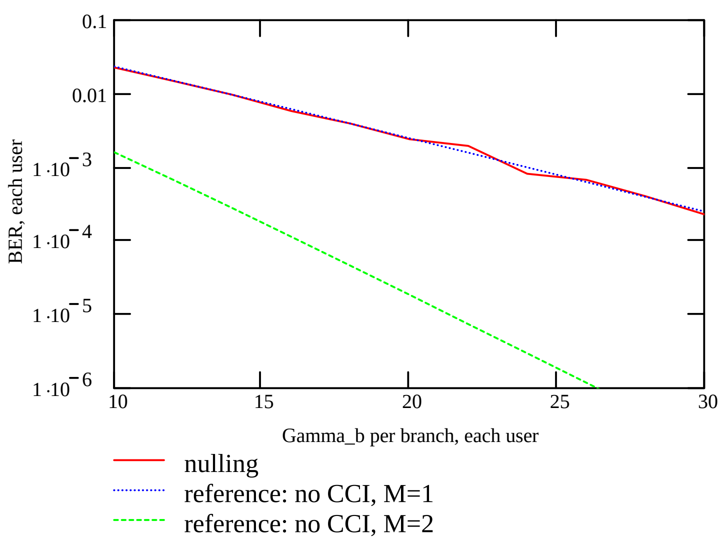
$k := 0..\ 10 \qquad \Gamma dB_k := 10 + 2\cdot k \qquad \Gamma_k := \mathrm{nat}(\Gamma dB_k) \qquad \mathrm{Nsim} := 10000$

$Pe_k := \mathrm{BER}(\Gamma_k, \mathrm{Nsim}) \qquad \mathrm{WRITEPRN}(\text{"Penull.txt"}) := Pe$

from 10 to 30 dB

$Pe := \mathrm{READPRN}(\text{"Penull.txt"}) \qquad Pe :=$

	0
0	0.02
1	0.01



K=2 , M=2, null steered, equipower users

These results are for Λ=0 dB! Equipower users.

The performance is identical to that of a single user with only one antenna (see reference curve) because zero forcing costs an order of diversity to null the other user.

Each interferer is invisible to the other.

Simulation of Adaptive MMSE Combining to Suppress Interferers

In this simulation, we combine the antennas according to a minimum mean squared error criterion - minimising the sum of cochannel interference (CCI) and noise variances. The interferers are suppressed, but not nulled out.

antennas | sources (mobiles)

$M := 2$ | $K := 2$

To keep it simple, we'll have one desired signal (signal 0) and K-1 interferers, all of equal power but weaker than the desired signal, with C/I $\Lambda > 1$. To set the scale, assume that amplitude $A_0 = 1$, and that Γ is the SNR per symbol of user 0. Also set the channel complex gain variances $\sigma_g{}^2 = 1$. The **A** matrix (diagonal matrix of signal amplitudes) is then prepared by

$$\mathrm{Agen}(\Lambda) := \left| \begin{array}{l} a_0 \leftarrow 1 \\ \text{for } k \in 1..K-1 \\ \quad a_k \leftarrow \dfrac{a_0}{\sqrt{\Lambda}} \\ \mathrm{diag}(a) \end{array} \right.$$

the noise variance is $\sigma_n{}^2 = \dfrac{1}{\Gamma}$

and the SNRs of the interferers are all

$$\Gamma_i = \frac{\Gamma}{\Lambda}$$

channel gain matrix generation, a column for each user

$$\mathrm{Ggen}(\mathrm{dummy}) := \left| \begin{array}{l} \text{for } i \in 0..M-1 \\ \quad \text{for } n \in 0..K-1 \\ \quad\quad G_{i,n} \leftarrow \mathrm{cgauss}(i) \\ G \end{array} \right.$$

the noise variances after weighting:

$$\mathrm{var}_\eta(\mathbf{W}, \Gamma) := \left| \begin{array}{l} \mathbf{Temp} \leftarrow \dfrac{1}{\Gamma} \cdot \left(\overline{\mathbf{W}^T} \cdot \mathbf{W} \right) \\ \text{for } k \in 0..K-1 \\ \quad \mathbf{var}_k \leftarrow \mathbf{Temp}_{k,k} \\ \mathbf{var} \end{array} \right.$$

slicing for decisions:

$$\mathrm{sgn}(x) := \mathrm{if}(x > 0, 1, -1)$$

useful vector

$k := 0..K-1$

$\mathbf{zero}_k := 0$

length-K binary data vectors can be indexed by integers; here we create the K-tuple from the index:

$$\mathrm{LSB}(i) := 2 \cdot \mathrm{mod}(i, 2) - 1$$

$$\mathrm{cvect}(i, K) := \left| \begin{array}{l} \text{for } m \in 0..K-1 \\ \quad \left| \begin{array}{l} \mathrm{temp} \leftarrow \mathrm{floor}\left(\dfrac{i}{2^m} \right) \\ \mathbf{c}_m \leftarrow \mathrm{LSB}(\mathrm{temp}) \end{array} \right. \\ \mathbf{c} \end{array} \right.$$

calculate the optimum weights (gain vectors and weight vectors are in columns of **G** and **W**

$$\mathrm{weights}(\mathbf{G}, \Gamma, \Lambda, \mathbf{A}) := \left| \begin{array}{l} \mathbf{R} \leftarrow \Gamma \cdot \mathbf{G}^{\langle 0 \rangle} \cdot \overline{\mathbf{G}^{\langle 0 \rangle}}^T \\ \text{for } k \in 1..M-1 \\ \quad \mathbf{R} \leftarrow \mathbf{R} + \dfrac{\Gamma}{\Lambda} \cdot \mathbf{G}^{\langle k \rangle} \cdot \overline{\mathbf{G}^{\langle k \rangle}}^T \\ \mathbf{R} \leftarrow \mathbf{R} + \mathrm{identity}(M) \\ \mathbf{W} \leftarrow \mathbf{R}^{-1} \cdot \mathbf{G} \cdot \mathbf{A} \cdot \Gamma \\ \mathbf{W} \end{array} \right.$$

The simulation performs a Monte Carlo average over the fading and data ensembles, but the noise ensemble average is handled analytically. Further efficiency is obtained by nesting the average over data (which is quite fast) inside the average over fading (which is computationally demanding because of weight calculations).

$$
\mathrm{BERmmse}\left(\Gamma,\Lambda,N_{sim}\right) :=
\left|
\begin{array}{l}
\mathbf{sum} \leftarrow \mathbf{zero} \\
\mathbf{A} \leftarrow \mathrm{Agen}(\Lambda) \\
\text{for } n \in 1..N_{sim} \\
\quad \left|
\begin{array}{l}
\mathbf{G} \leftarrow \mathrm{Ggen}(n) \\
\mathbf{W} \leftarrow \mathrm{weights}(\mathbf{G},\Gamma,\Lambda,\mathbf{A}) \\
\mathbf{H} \leftarrow \mathbf{A}\cdot\overline{\mathbf{G}^{T}}\cdot\mathbf{W} \\
\eta\mathrm{var} \leftarrow \mathrm{var}_{\eta}(\mathbf{W},\Gamma) \\
\sigma_{\eta} \leftarrow \overrightarrow{\sqrt{\eta\mathrm{var}}} \\
\mathbf{sum2} \leftarrow \mathbf{zero} \\
\text{for } i \in 0..2^{K}-1 \\
\quad \left|
\begin{array}{l}
\mathbf{c} \leftarrow \mathrm{cvect}(i,K) \\
\text{for } k \in 0..K-1 \\
\quad \mathbf{p}_{k} \leftarrow Q\left[\mathrm{Re}\left[\dfrac{\left(\overline{\mathbf{H}^{\langle k\rangle}}^{T}\cdot\mathbf{c}\right)_{0}\cdot\mathrm{sgn}\left(\mathbf{c}_{k}\right)}{\sigma_{\eta_{k}}}\right]\right] \\
\mathbf{sum2} \leftarrow \mathbf{sum2}+\mathbf{p}
\end{array}
\right. \\
\mathbf{sum} \leftarrow \mathbf{sum}+\dfrac{\mathbf{sum2}}{2^{K}}
\end{array}
\right. \\
\dfrac{\mathbf{sum}}{N_{sim}}
\end{array}
\right.
$$

Below is a work area for running MMSE simulations and saving the results. A graphical display is given on the page after next:

from 10 to 30 dB

$k := 0..10$ $\quad \Gamma dB2_{k} := 10+2\cdot k$ $\quad \Gamma 2_{k} := \mathrm{nat}\left(\Gamma dB2_{k}\right)$ $\quad N_{sim} := 200000$ $\quad \Lambda := \mathrm{nat}(0)$

$\mathrm{Pem}^{\langle k\rangle} := \mathrm{BERmmse}\left(\Gamma 2_{k},\Lambda,\mathrm{Nsim}\right)$ $\quad \mathrm{Pem} := \mathrm{augment}\left(\Gamma dB2,\mathrm{Pem}^{T}\right)$

Pem =

Save the results of the run. Filename code: Pmse K M ΛdB.txt, file contents: ΓdB, Pe stronger, Pe weaker.

WRITEPRN("Pmse2200.txt") := Pem

For the plots on the next page, read in saved simulation results:

Pem00 := READPRN("Pmse2200.txt")   Pem10 := READPRN("Pmse2210.txt")   Pem20 := READPRN("Pmse2220.txt")

Pem30 := READPRN("Pmse2230.txt")

File reads don't work on a website, so they are replaced here with one-time table imports:

Pem00 :=

	0
0	10

Pem10 :=

	0
0	10

Pem20 :=

	0
0	10

Pem30 :=

	0
0	10

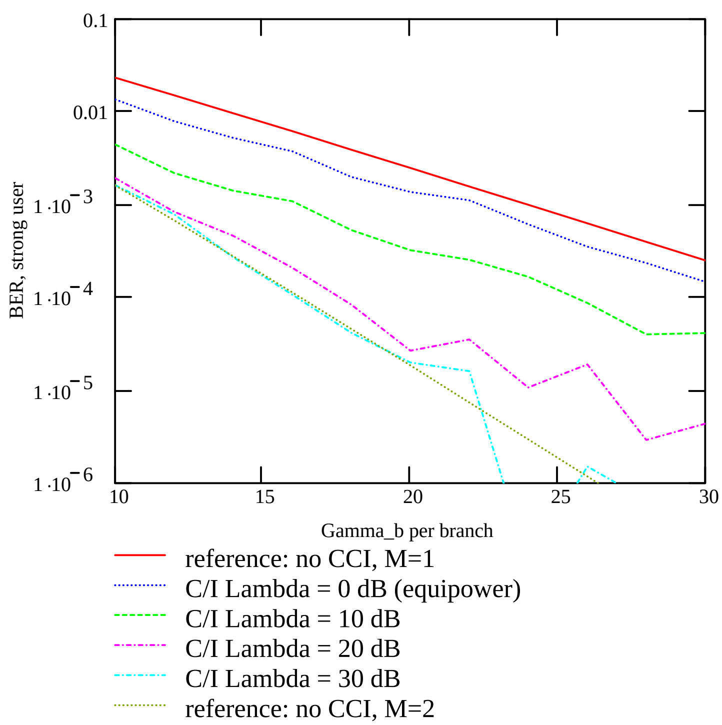
$Pedes00 := Pem00^{\langle 1 \rangle}$ $Pedes10 := Pem10^{\langle 1 \rangle}$ $Pedes20 := Pem20^{\langle 1 \rangle}$ $Pedes30 := Pem30^{\langle 1 \rangle}$

$Pedes00lo := Pem00^{\langle 2 \rangle}$ $Pedes10lo := Pem10^{\langle 2 \rangle}$ $Pedes20lo := Pem20^{\langle 2 \rangle}$ $Pedes30lo := Pem30^{\langle 2 \rangle}$

$\Gamma 10 := \Gamma dB - 10$ $\Gamma 20 := \Gamma dB - 20$ $\Gamma 30 := \Gamma dB - 30$

|| Skip down to the next page to see displayed results
||
||
\/

And here are some results from the files:



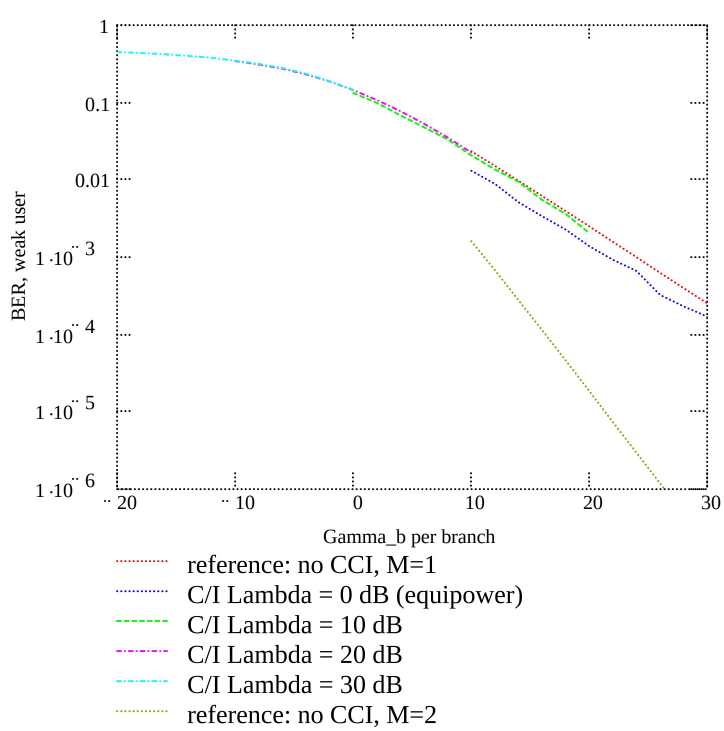
K=2 , M=2, MMSE, Stronger User

Notes:

* operates like max ratio when CCI is much less than noise, hence roughly dual diversity

* when CCI dominates noise, operates more like null steering, hence roughly single diversity (no diversity), as seen in the slope - but there's a still a SNR benefit compared to pure nulling

* when users are equipower, then MMSE is a little better than zero forcing (the reference curve) in the SNR range shown.



K=2 , M=2, MMSE, Weaker User

Notes:

* All of the MMSE curves for the weaker user lie on top of the single antenna (M=1) reference curve. This shows that the solution is always very close to the zero forcing (nulling) solution, and it lost an order of diversity.