6.B SIMULATION OF MUD WITHOUT SIGNATURES

1. INTRODUCTION

The simulation in this worksheet compares the performance of four detectors of interfering narrowband signals in Rayleigh fading. The lineup: single user detector; zero forcing (ZF), minimum mean squared error (MMSE) and maximum likelihood (ML). All detectors work on the same received signals, for more accurate comparisons.

Notation is as follows:

K number of users M number of antennas

b length-K vector of bits from all users

A KxK diagonal array of amplitudes of all users

C MxK array of gains; Gaussian, variance 1/2; column k $\mathbf{C}^{\langle k \rangle}$ is for user k.

n length-M array of noises on the antennas; unit variance (i.e. normalised so N_o=1). With the noise normalisation, we have each amplitude as $A_k = \sqrt{2 \cdot \Gamma_k}$, where Γ_k is the user-k SNR.

Several useful procedures are in the Appendix.

2. One-Shot Test of Simulation

In this section, we build up the simulation, to see how it fits together. In the next section, we condense to a single procedure that does many, many trials. Highlighted regions are parameter input.

$K := 2$ $M := 2$ users and antennas, resp. $\mathbf{I_K} := \text{identity}(K)$ $\mathbf{I_M} := \text{identity}(M)$

$\mathbf{b} := \text{datavec}(K)$ transmitted data (for another realisation, click on equation and press F9)

$\Lambda_{dB} := 0$ $\Lambda := \text{nat}(\Lambda_{dB})$ ratio of user 1 power to user 0 power (SIR)

$\Gamma_{dB_0} := 3$ $\Gamma_0 := \text{nat}(\Gamma_{dB_0})$ $\Gamma_1 := \Gamma_0 \cdot \Lambda$ the two SNRs

$\mathbf{A} := \text{diag}\left(\overrightarrow{\sqrt{2 \cdot \Gamma}}\right)$ make the diagonal matrix of amplitudes (sqrt computed component by component)

$\mathbf{C} := \frac{1}{\sqrt{2}} \cdot \text{gaussarray}(M, K)$ generate the matrix of channel gains, all i.i.d., variance 1/2 (for another realisation, click on equation and press F9)

$\mathbf{F} := \mathbf{C}\cdot\mathbf{A}$ more concise notation

$\mathbf{n} := \text{gaussarray}(M, 1)$ the vector of noises, variance 1 (for another realisation, click on equation and press F9)

$\mathbf{y} := \mathbf{F}\cdot\mathbf{b} + \mathbf{n}$ vector of matched filter outputs

$z := \overline{\mathbf{F}^{T}}\cdot\mathbf{y}$ sufficient statistics (basically max ratio combining)

Single User Detector

$\mathbf{b_s} := \overrightarrow{\text{sgn}(z)}$ pretends other users aren't there

Zero Forcing

$\text{bhat} := \left(\overline{\mathbf{F}^{T}}\cdot\mathbf{F}\right)^{-1}\cdot z$ pseudoinverse on $\mathbf{y}$ $\mathbf{b_{zf}} := \overrightarrow{\text{sgn}(\text{bhat})}$

MMSE

$\text{bhat} := \overline{\mathbf{F}^{T}}\cdot\left(\mathbf{F}\cdot\overline{\mathbf{F}^{T}} + 2\cdot\mathbf{I_M}\right)^{-1}\cdot\mathbf{y}$ or $\text{bhat} := \left(\overline{\mathbf{F}^{T}}\cdot\mathbf{F} + 2\cdot\mathbf{I_K}\right)^{-1}\cdot z$ different, but equivalent, forms

$\mathbf{b_{mmse}} := \overrightarrow{\text{sgn}(\text{bhat})}$

ML

generate matrix of all possible data patterns

$i := 0 .. 2^{K} - 1$ $\mathbf{B}^{\langle i\rangle} := \text{int\_to\_bvec}(i, K)$

calculate metric of all patterns, then find index of minimum metric

$\text{metric}_i := \left(\left|\mathbf{y} - \mathbf{C}\cdot\mathbf{A}\cdot\mathbf{B}^{\langle i\rangle}\right|\right)^2$ $\text{iopt} := \text{smallest}(\text{metric})_1$ $\mathbf{b_{ml}} := \mathbf{B}^{\langle \text{iopt}\rangle}$

Compare the decisions

$\mathbf{b_s} = \begin{pmatrix}-1\\1\end{pmatrix}$ $\mathbf{b_{zf}} = \begin{pmatrix}-1\\-1\end{pmatrix}$ $\mathbf{b_{mmse}} = \begin{pmatrix}-1\\-1\end{pmatrix}$ $\mathbf{b_{ml}} = \begin{pmatrix}-1\\-1\end{pmatrix}$

$\text{errs}(\mathbf{b_s}, \mathbf{b}) = \begin{pmatrix}0\\1\end{pmatrix}$ $\text{errs}(\mathbf{b_{zf}}, \mathbf{b}) = \begin{pmatrix}0\\0\end{pmatrix}$ $\text{errs}(\mathbf{b_{mmse}}, \mathbf{b}) = \begin{pmatrix}0\\0\end{pmatrix}$ $\text{errs}(\mathbf{b_{ml}}, \mathbf{b}) = \begin{pmatrix}0\\0\end{pmatrix}$

3. THE FULL SIMULATION

Here we put all the detectors head-to-head in the same environment. May the best detector win.

Here is one run. We'll do several of them further below.

$$\text{MUDfight}(K, M, \mathbf{A}, 1000) = \begin{pmatrix} 0.092 & 0.09 & 0.054 & 0.034 \\ 0.097 & 0.093 & 0.062 & 0.035 \end{pmatrix}$$

A column for each detector: single-user, ZF, MMSE and ML. Row 0 is user 0, row 1 is user 1.

$\text{MUDfight}(K, M, \mathbf{A}, N_{sim}) :=$

$\mathbf{I_K} \leftarrow \text{identity}(K)$

for $i \in 0 .. 2^K - 1$

$\mathbf{B}^{\langle i \rangle} \leftarrow \text{int\_to\_bvec}(i, K)$

$\mathbf{E} \leftarrow \text{zeros}(K, 4)$

for $n \in 1 .. N_{sim}$

$\mathbf{b} \leftarrow \text{datavec}(K)$

$\mathbf{C} \leftarrow \frac{1}{\sqrt{2}} \cdot \text{gaussarray}(M, K)$

$\mathbf{F} \leftarrow \mathbf{C} \cdot \mathbf{A}$

$\mathbf{y} \leftarrow \mathbf{F} \cdot \mathbf{b} + \text{gaussarray}(M, 1)$

$z \leftarrow \overline{\mathbf{F}^T} \cdot \mathbf{y}$

$\mathbf{b_s} \leftarrow \overrightarrow{\text{sgn}(z)}$

$\mathbf{E}^{\langle 0 \rangle} \leftarrow \mathbf{E}^{\langle 0 \rangle} + \text{errs}(\mathbf{b_s}, \mathbf{b})$

$\text{bhat} \leftarrow \left(\overline{\mathbf{F}^T} \cdot \mathbf{F}\right)^{-1} \cdot z$

$\mathbf{b_{zf}} \leftarrow \overrightarrow{\text{sgn}(\text{bhat})}$

$\mathbf{E}^{\langle 1 \rangle} \leftarrow \mathbf{E}^{\langle 1 \rangle} + \text{errs}(\mathbf{b_{zf}}, \mathbf{b})$

$\text{bhat} \leftarrow \left(\overline{\mathbf{F}^T} \cdot \mathbf{F} + 2 \cdot \mathbf{I_K}\right)^{-1} \cdot z$

$\mathbf{b_{mmse}} \leftarrow \overrightarrow{\text{sgn}(\text{bhat})}$

$\mathbf{E}^{\langle 2 \rangle} \leftarrow \mathbf{E}^{\langle 2 \rangle} + \text{errs}(\mathbf{b_{mmse}}, \mathbf{b})$

for $i \in 0 .. 2^K - 1$

$\text{metric}_i \leftarrow \left(\left|\mathbf{y} - \mathbf{F} \cdot \mathbf{B}^{\langle i \rangle}\right|\right)^2$

$\text{iopt} \leftarrow \text{smallest}(\text{metric})_1$

$\mathbf{b_{ml}} \leftarrow \mathbf{B}^{\langle \text{iopt} \rangle}$

$\mathbf{E}^{\langle 3 \rangle} \leftarrow \mathbf{E}^{\langle 3 \rangle} + \text{errs}(\mathbf{b_{ml}}, \mathbf{b})$

$\frac{\mathbf{E}}{N_{sim}}$

and this one loops through a sequence of SNR values:

$$\text{MUDcurves}(M, G\mathbf{0dB}, L\mathbf{dB}, N_{sim}) :=$$
$$K \leftarrow \text{length}(L\mathbf{dB})$$
$$\text{all} \leftarrow \text{zeros}(1, 8)$$
$$\text{for } i \in 0 .. \text{length}(G\mathbf{0dB}) - 1$$
$$\quad G\mathbf{dB} \leftarrow G\mathbf{0dB}_i \cdot \text{ones}(K, 1) + L\mathbf{dB}$$
$$\quad A \leftarrow \overrightarrow{\sqrt{2 \cdot \text{nat}(G\mathbf{dB})}}$$
$$\quad \mathbf{A} \leftarrow \text{diag}(A)$$
$$\quad \mathbf{BERs} \leftarrow \text{MUDfight}(K, M, \mathbf{A}, N_{sim})$$
$$\quad \text{keep} \leftarrow \text{augment}(G\mathbf{dB}, K \cdot \text{ones}(K, 1), M \cdot \text{ones}(K, 1))$$
$$\quad \text{keep} \leftarrow \text{augment}(\text{keep}, N_{sim} \cdot \text{ones}(K, 1), \mathbf{BERs})$$
$$\quad \text{all} \leftarrow \text{stack}(\text{all}, \text{keep})$$
$$\text{all}$$

Next, we run and save the simulation for BERs at a sequence of SNR values. This group is for **equipower users.** It's in the area below that we harvest the results of our work in building the simulation.

$K := 2$ $M := 2$ users and antennas, resp.

$k := 0 .. K - 1$ $L\mathbf{dB}_k := 0$ ratio (in dB) of user i power to user 0 power

$$G\mathbf{0dB} := \begin{pmatrix} 5 \\ 10 \\ 15 \\ 20 \end{pmatrix} \qquad N_{sim} := 500000 \qquad L\mathbf{dB} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

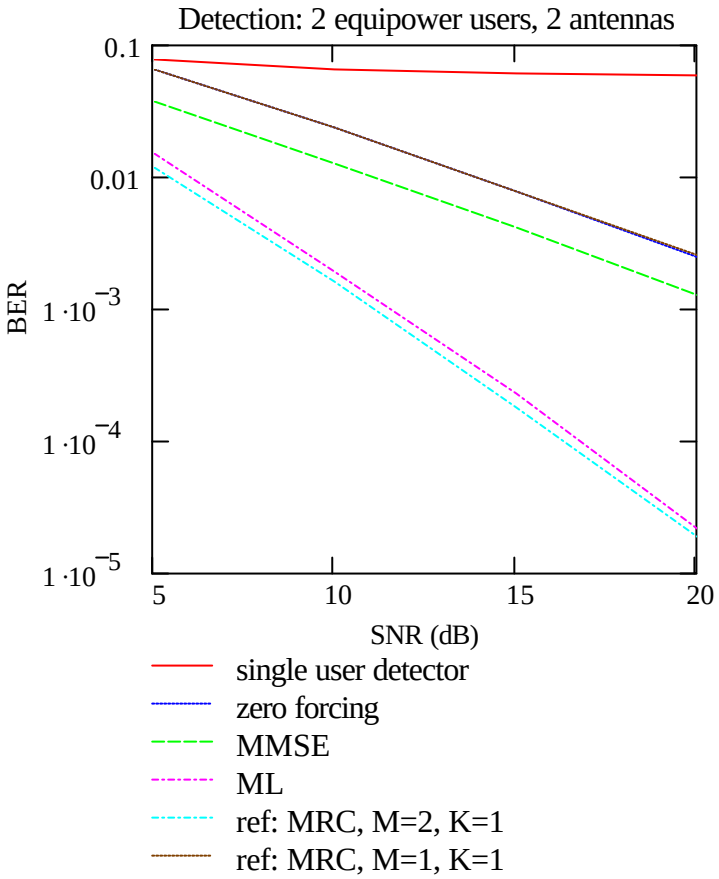
$\text{results1} := \text{MUDcurves}(M, G\mathbf{0dB}, L\mathbf{dB}, N_{sim})$

$\text{results1} := \text{submatrix}(\text{results1}, 1, \text{rows}(\text{results1}) - 1, 0, \text{cols}(\text{results1}) - 1)$ trim off the row of zeros

$\text{WRITEPRN}(\text{"equipowerdet.txt"}) := \text{results1}$ $\text{temp} := \text{READPRN}(\text{"equipowerdet.txt"})^T$

average the rows, since equipower, same BER $i := 0 .. \frac{\text{cols}(\text{temp})}{2} - 1$

$\text{BER}^{\langle i \rangle} := 0.5 \cdot (\text{temp}^{\langle 2 \cdot i \rangle} + \text{temp}^{\langle 2 \cdot i + 1 \rangle})$ $\text{BER} := \text{BER}^T$



Trials of 10^6 bits for each point.

With equipower users, the single-user detector is useless. As expected, MMSE is a little better than ZF (caution - this is for equipower users), but both have lost an order of diversity. Also as expected, ML is best and retains the dual diversity. It seems unaffected by the interference.

4. NEAR-FAR EFFECTS

In this section, we examine the effects of dissimilar power levels among users. User 1 is a few dB below user 0.

$K := 2$ $M := 2$ users and antennas, resp.

$L\mathbf{dB}_0 := 0$ $L\mathbf{dB}_1 := -10$ ratio (in dB) of user i power to user 0 power

$$G\mathbf{0dB} := \begin{pmatrix} 5 \\ 10 \\ 15 \\ 20 \\ 25 \end{pmatrix} \qquad N_{sim} := 500000$$

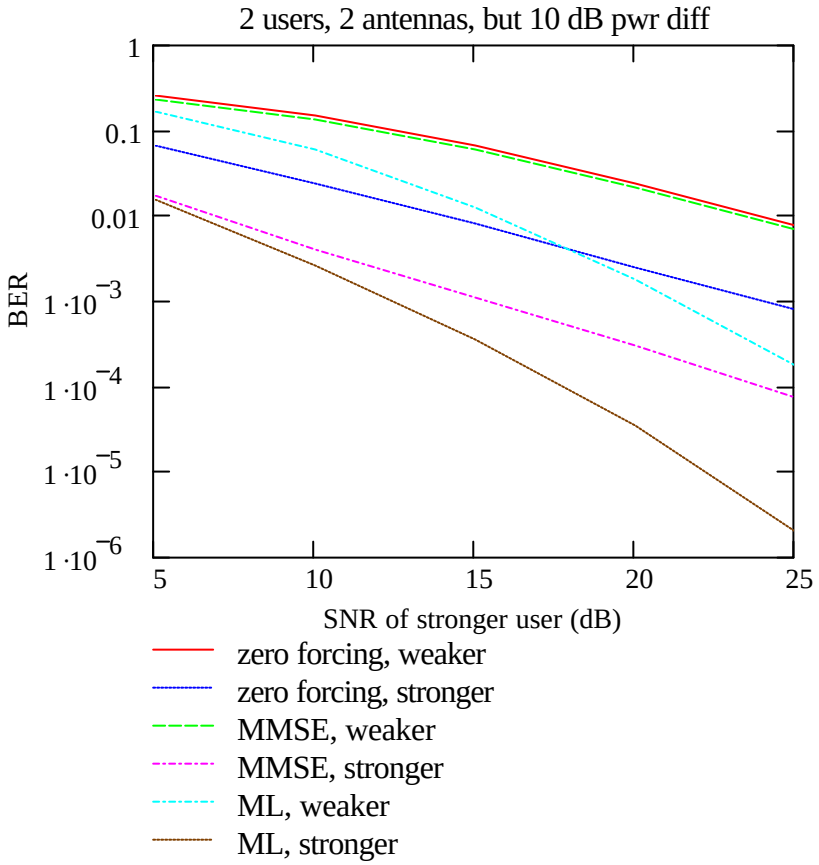
$results2 := MUDcurves(M, G\mathbf{0dB}, L\mathbf{dB}, N_{sim})$

$results2 := submatrix(results2, 1, rows(results2) - 1, 0, cols(results2) - 1)$ trim off the row of zeros

$WRITEPRN("pdiff10.txt") := results2$

As a first presentation of the results, we'll plot against the SNR of the stronger user. Ignore the single-user detector. Trials of 500,000 bits for each point.

BER := READPRN("pdiff10.txt") r := rows(BER) i := 0, 2 .. rows(BER) − 2



Observations

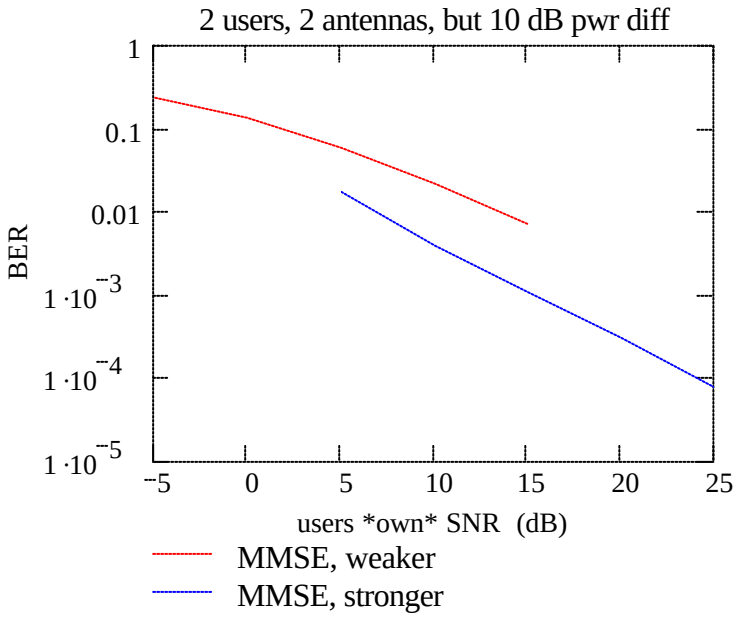
* ZF is equivalent to MRC with one user and one antenna. Weak shifted right by exactly 10 dB with respect to strong user.

* Weak MMSE user has same error rate as weak ZF, since MMSE acts much like ZF for it.

* Strong MMSE user is like ML, at SNR region where weak user has power comparable to noise; asymptotic single diversity, but still much better than strong ZF user.

* ML has dual diversity for both users; weak curve shifted right with respect to stronger user.

What if we plot against the user's *own* SNR? Should be revealing. The first is for MMSE.



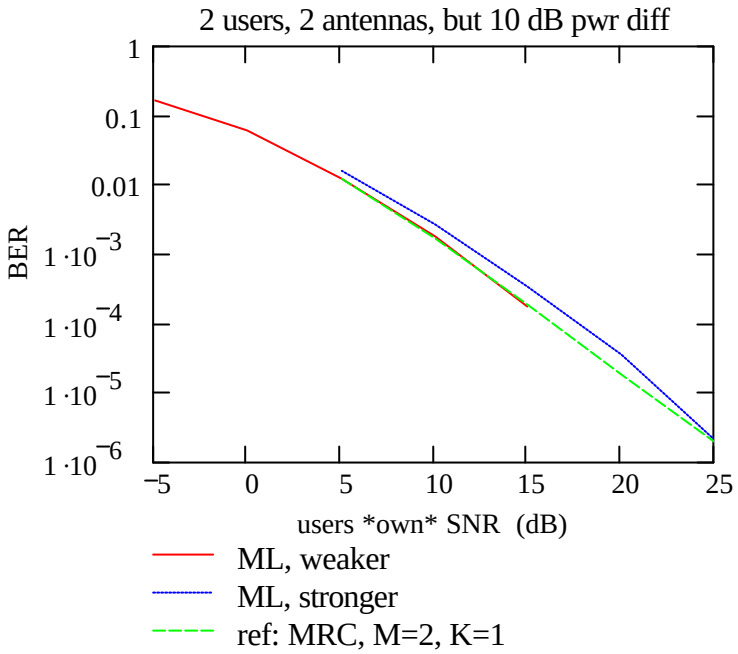
Observations

* Plotting against own SNR removes the obvious issue that weak users would have poorer BER, anyway.

* In the overlap area, the stronger user gets the better of it, even after accounting for power disparity. Weaker user is disadvantaged by presence of stronger.

* Conclusion: MMSE is only partially effective against near-far differences; equivalently, power disparities.

And our third plot shows ML detection, again when plotted against *own* SNR.



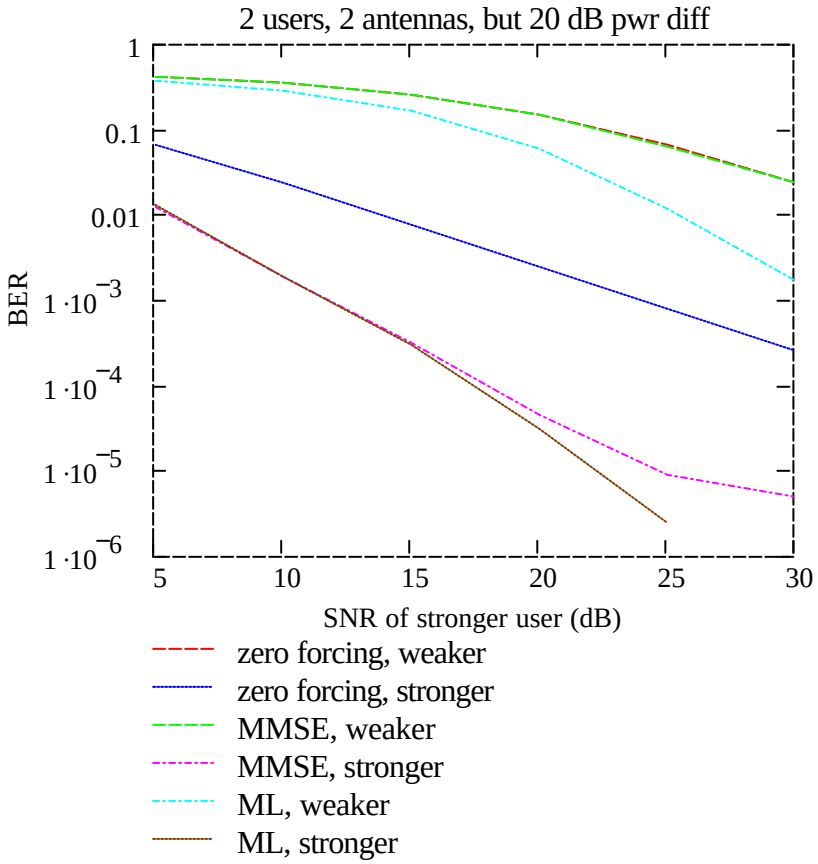
Observations

* ML is much more robust in face of near-far or power disparities.

* In constrast to MMSE, it is the *weaker* user who gets a little more benefit from ML-MUD. But remember the assumption of perfect CSI. Hmmm.

Do things change if there is a 20 dB power difference? Let's have a look. This time 1.2 million trials.

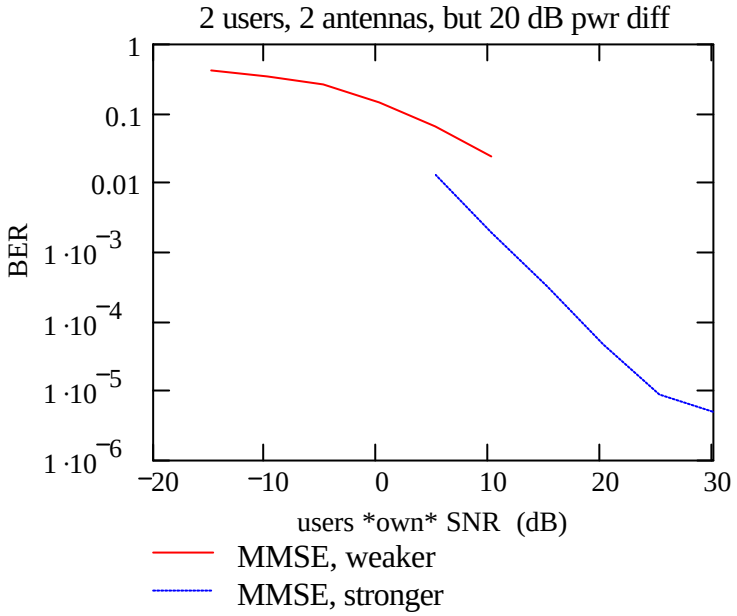
$BER := READPRN("pdiff20.txt")$ $r := rows(BER)$ $i := 0, 2 .. rows(BER) - 2$



Observations

* For the stronger user, MMSE is as good as ML over much of the range; however, this is because it is, in effect, a single user system, and both operate like MRC. With more than one strong user, MMSE would again lose orders of diversity.

Next, we plot against the user's *own* SNR, as we did before. The first is for MMSE.

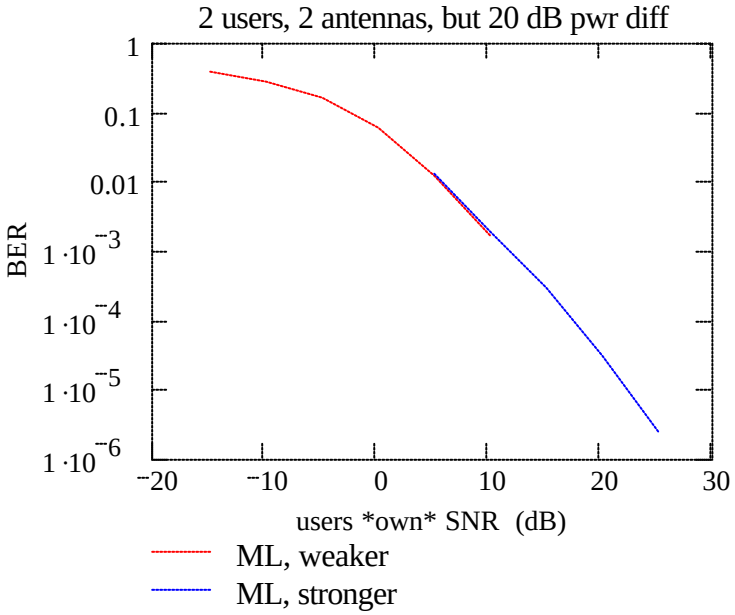


Observations

* In the overlap area, the stronger user gets the better of it, even after accounting for power disparity. Weaker user is disadvantaged by presence of stronger.

* Conclusion: MMSE is only partially effective against near-far differences; equivalently, power disparities.

And our last plot shows ML detection, again when plotted against *own* SNR.



Observations

* ML is much more robust in face of near-far or power disparities.

* In ML-MUD, the users track the same curve. MUD has not disadvantaged either user.

APPENDIX A: USEFUL FUNCTIONS

This appendix contains a few "housekeeping" functions that simplify the simulation.

Basic definitions and functions

$$\text{smaller}(x,y) \equiv \text{if}(x < y, x, y) \qquad \text{larger}(x,y) \equiv \text{if}(x > y, x, y)$$

$$\text{dB}(x) \equiv \text{if}\left(x > 10^{-14}, 10\cdot\log(x), -140\right) \qquad \text{nat}(x) \equiv 10^{0.1\cdot x}$$

$$\text{sgn}(x) \equiv \text{if}(\text{Re}(x) \geq 0, 1, -1)$$

Manipulations of binary data vectors

Hamming error vector

$$\text{errs}(\mathbf{b},\mathbf{c}) \equiv \left|\begin{array}{l} \mathbf{e} \leftarrow \mathbf{b} - \mathbf{c} \\ \text{for } i \in 0..\text{length}(\mathbf{e}) - 1 \\ \quad \mathbf{e}_i \leftarrow \text{if}(\mathbf{e}_i = 0, 0, 1) \\ \mathbf{e} \end{array}\right.$$

Conversions between integers and binary data vectors (of +1,-1)

$$\text{int\_to\_bvec}(n,N) \equiv \left|\begin{array}{l} i \leftarrow 0 \\ \text{while } i < N \\ \quad \left|\begin{array}{l} \text{rem} \leftarrow \text{mod}(n,2) \\ b_i \leftarrow \text{if}(\text{rem} = 1, 1, -1) \\ n \leftarrow \text{floor}\left(\frac{n}{2}\right) \\ i \leftarrow i + 1 \end{array}\right. \\ b \end{array}\right.$$

$$\text{bvec\_to\_int}(b) \equiv \left|\begin{array}{l} \text{sum} \leftarrow 0 \\ \text{for } i \in 0..\text{rows}(b) - 1 \\ \quad \text{sum} \leftarrow \text{sum} + 2^{i-1}\cdot(b_i + 1) \\ \text{sum} \end{array}\right.$$

Generate random vectors and arrays

Generate random data vector

$$\text{datavec}(N) \equiv \text{int\_to\_bvec}\left(\text{floor}\left(\text{rnd}\left(2^N\right)\right), N\right)$$

Generate array (N_r x N_c) of variance 1 complex Gaussian

$$\text{cgauss}(x) \equiv \sqrt{-2\cdot\ln(\text{rnd}(1))}\cdot\exp\left(j\cdot\text{rnd}\left(2\cdot\pi\right)\right)$$ unit variance complex Gaussian

$$\text{gaussarray}\left(N_r, N_c\right) \equiv \left|\begin{array}{l} \text{for } ir \in 0..N_r - 1 \\ \quad \text{for } ic \in 0..N_c - 1 \\ \quad\quad A_{ir,ic} \leftarrow \text{cgauss}(ir) \\ A \end{array}\right.$$

Array manipulations

Make an all-zeros array

$$\text{zeros}\left(N_r, N_c\right) \equiv \left|\begin{array}{l} \text{for } ir \in 0..N_r - 1 \\ \quad \text{for } ic \in 0..N_c - 1 \\ \quad\quad A_{ir,ic} \leftarrow 0 \\ A \end{array}\right.$$

Make an all-ones array

$$\text{ones}\left(N_r, N_c\right) \equiv \left|\begin{array}{l} \text{for } ir \in 0..N_r - 1 \\ \quad \text{for } ic \in 0..N_c - 1 \\ \quad\quad A_{ir,ic} \leftarrow 1 \\ A \end{array}\right.$$

find smallest/largest in an array; return the value and its index:

$$\text{smallest}(x) \equiv \left|\begin{array}{l} \text{small} \leftarrow 10^{15} \\ \text{for } i \in 0..\text{rows}(x) - 1 \\ \quad \text{if } x_i < \text{small} \\ \quad\quad \left|\begin{array}{l} \text{small} \leftarrow x_i \\ \text{i\_small} \leftarrow i \end{array}\right. \\ (\text{small}\quad \text{i\_small})^T \end{array}\right.$$

$$\text{largest}(x) \equiv \left|\begin{array}{l} \text{large} \leftarrow -10^{15} \\ \text{for } i \in 0..\text{rows}(x) - 1 \\ \quad \text{if } x_i > \text{large} \\ \quad\quad \left|\begin{array}{l} \text{large} \leftarrow x_i \\ \text{i\_large} \leftarrow i \end{array}\right. \\ (\text{large}\quad \text{i\_large})^T \end{array}\right.$$