Ultra Wide Band Radio Fundamentals

MUI models for IR-UWB

DIET Department

SAPIENZA
UNIVERSITÀ DI ROMA

Outline

SGA

- Sigma effect
- BER
- BER floor

PC

Theory vs. simulation

Outline

SGA

- Sigma effect
- BER
- BER floor

PC

Theory vs. simulation

Remind:

decision variable

$$Z = \int_{\hat{o}}^{N_S T_S + \hat{o}} r(t) m(t - \hat{o}) dt$$

received signal

$$r(t) = s_{RX}(t) + s_{mui}(t) + n(t)$$

$$Z = Z_u + Z_{MUI} + Z_n$$

the MUI term

$$Z_{mui} = \sum_{i=1}^{N_i} \sqrt{E^{(i)}} \sum_{j=0}^{N_S} \int_{-\infty}^{\infty} p_0(t - \theta_j)(p_0(t) - p_0(t - \varepsilon)) dt$$

Z_{mui} is a random variable with mean zero and variance:

$$\sigma_{mui}^2 = \sum_{i=1}^{N_i} E^{(i)} N_S \frac{1}{T_S} \underbrace{\int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} p_0(t - \tau)(p_0(t) - p_0(t - \varepsilon)) dt \right)^2 d\tau}_{\sigma_M^2}$$

σ_M^2 constant term which depends on both pulse shape and PPM shift

$$\sigma_{mui}^2 = \sum_{i=1}^{N_i} E^{(i)} N_S \frac{\sigma_M^2}{T_S} = \frac{N_S}{T_S} \sigma_M^2 \sum_{i=1}^{N_i} E^{(i)}$$

Sigma effect

The power of the MUI component of the decision variable is:

$$\sigma_{MUI}^2 = \frac{N_s}{Ts}\sigma_M^2 \sum_{i=1}^{N_i} E^{(i)}$$

If power control:

$$E^{(i)} = E_{RX} \; \forall \; i$$

where:

N_i – number of interferers

$E^{(i)}$ – energy of the i-th interferer

and:

$$\sigma_M^2 = \int_0^{Ts} \left\{ \int_0^{2T_M} p_0(t-\tau)\left[p_0(t) - p_0(t-\varepsilon)\right]dt \right\}^2 d\tau$$

If $\varepsilon > T_M$ ⇒ $\sigma_M^2 = \int_{-T_M}^{T_M} \left(R_0(\tau)\right)^2 d\tau$

Orthogonal pulses

Sigma effect: simulation

GOAL: show the effect of ε on σ_M^2 in PPM and PAM modulations

Write the function

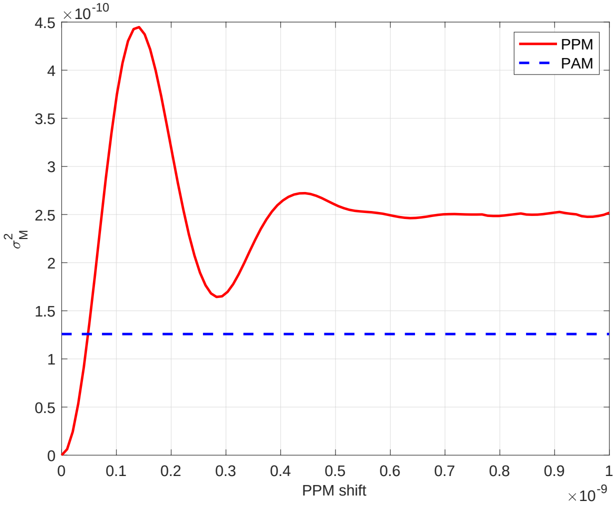
```
sm2 = sm2_mod(mod, pulse, dPPM, fc)
```

to highlight the *asymptotic* difference of σ_M^2 between PPM and PAM

Settings

- `mod`: 1 - PPM, 2 - PAM
- `pulse`: 2nd derivative of Gaussian pulse with τ=0.25 ns and T_M= 1 ns
- `dPPM`: PPM shift ε in [0,1] ns
- `fc`: sampling frequency: 100 GHz

Sigma effect: result



Outline

SGA

- Sigma effect
- BER
- BER floor

PC

Theory vs. simulation

Remind: The Standard Gaussian Approximation

$$\Pr_b = Prob\left(N_S\sqrt{E_{RX}}\left(1-R_0(\varepsilon)\right)+Z_{mui}+Z_n<0\right)$$

Gaussian random variable with mean zero and variance

$$\Pr_b = \frac{1}{2}erfc\left(\sqrt{\frac{1}{2}SNR_{spec}}\right)$$

$$SNR_{spec} = \frac{|Z_u|^2}{\sigma_{mui}^2+\sigma_n^2} = \left(\left(\frac{|Z_u|^2}{\sigma_n^2}\right)^{-1}+\left(\frac{|Z_u|^2}{\sigma_{mui}^2}\right)^{-1}\right)^{-1} = \left((SNR_n)^{-1}+(SNR_{mui})^{-1}\right)^{-1}$$

$$SNR_n = \frac{N_S^2E_{RX}\left(1-R_0(\varepsilon)\right)^2}{N_S\mathcal{N}_0\left(1-R_0(\varepsilon)\right)} = N_S\frac{E_{RX}}{\mathcal{N}_0}\left(1-R_0(\varepsilon)\right) = \frac{E_b}{\mathcal{N}_0}\left(1-R_0(\varepsilon)\right)$$

$$SNR_{mui} = \frac{N_S^2E_{RX}\left(1-R_0(\varepsilon)\right)^2}{\frac{N_S}{T_S}\sigma_M^2\sum_{i=1}^{N_i}E^{(i)}} = \frac{T_SN_S\left(1-R_0(\varepsilon)\right)^2}{\sigma_M^2\sum_{i=1}^{N_i}\frac{E^{(i)}}{E_{RX}}} = \frac{\left(1-R_0(\varepsilon)\right)^2}{R_b\sigma_M^2\sum_{i=1}^{N_i}\frac{E^{(i)}}{E_{RX}}}$$

BER (SGA)

GOAL: evaluate the BER for PPM and PAM according to SGA for different numbers of interferers

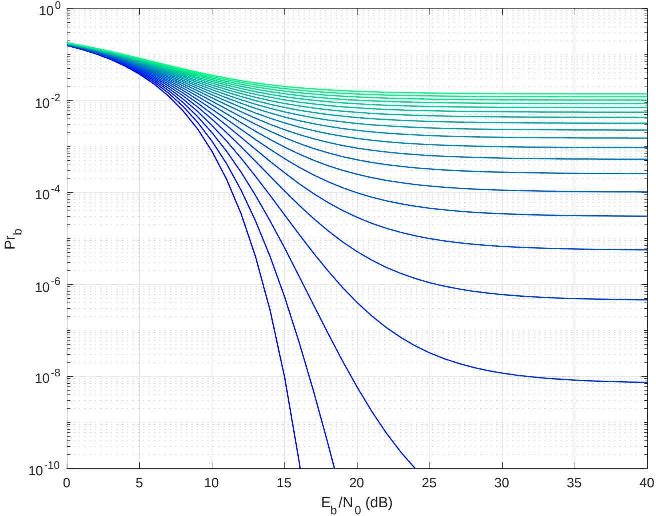
Write the function **MUIBER** defined as:

```
BER = MUIBER(mod,EbN0,erx0,erxMUI,pulse,Rb,dPPM,fc,gamma_r)
```

Settings

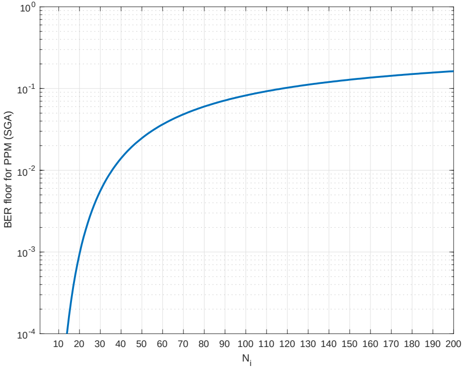
- `mod`: 1 - PPM, 2 - PAM
- `EbN0`: Eb/N0 in dB, [0, 40] dB
- `erx0`: normalized rx energy per pulse of the useful signal, 1 J
- `erxMUI`: vector of normalized rx energies per pulse of interferers, 1 J
- `Rb`: data rate, 20 Mb/s
- `dPPM`: 0.5 ns
- `gamma_r`: $\gamma_r = \frac{T_s}{T_b} N_s$, 1
- Ni (length of `erxMUI`): number of interferers, 1:2:39

Example: PPM



BER floor (SGA)

- Draw a plot showing the BER floor as a function of N_i for both modulation schemes, for N_i=1:200
- Example: PPM



- Compare the floor for PPM vs. PAM

Outline

SGA

- Sigma effect
- BER
- BER floor

PC

Theory vs. simulation

Reminder: Pulse Collision interference model

Assumptions:

- Asynchronous network $\Longrightarrow$ packet arrival is a Poisson process
- Pulse arrival is ALSO a Poisson process
- Collision probability is thus:

$$Pr_{PC} = 1 - e^{-2(N_u - 1)\frac{T_M}{T_s}}$$

- If pulse collision occurs then receiver decides randomly:

$$Pr_{PE} = 0.5 Pr_{PC}$$

- BER and PER (for a packet of length L, no error protection) are:

$$Pr_b = \sum_{i=\lceil \frac{N_s}{2} \rceil}^{N_s} \binom{N_s}{i} (Pr_{PE})^i (1 - Pr_{PE})^{N_s - i}$$

$$PER = 1 - (1 - Pr_b)^L$$

Reminder: Pulse Collision interference model

- Note that the PC model assumes that a collision occurs if two pulses overlap by even a small amount
- This may lead to an overestimation of the effect of interference
- This issue can be mitigated by introducing the concept of ***Effective Pulse Duration*** (EPD)
- The EPD is defined as the duration that contains a given percentage of the energy of the pulse
- EPD is then used in place of T_M in the estimation of the collision probability

Pulse Collision model: simulation time!

You will need to write two functions:

```
[epulse, EPD] = effpulse(pulse, fc, pE)
```

Returns the EPD that contains a percentage `pE` of the total energy of the pulse

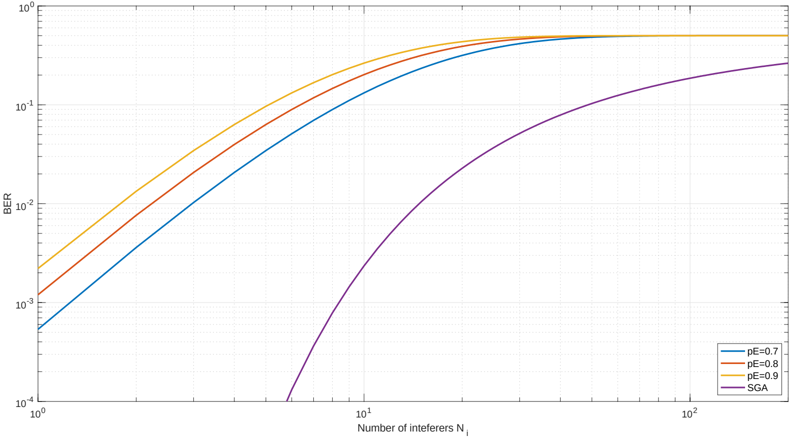
```
BER_PC= prbcol(Nu, Ns, EPD, Ts)
```

Implements the BER formula recalled in the previous slide

Settings

```
Nu  = 1:200             Tm = 1e-9
Ns  = 5                 fc = 50e9
Ts  = 3e-9              pE = 0.9, 0.8, 0.7
tau = 0.2e-9
```

Pulse Collision model: simulation time!



Outline

SGA

- Sigma effect
- BER
- BER floor

PC

Theory vs. simulation

BER: theory vs. simulation

GOAL: compare simulations with theory (SGA and PC models)

Procedure:

1. Generate the useful signal and N_i interfering signals with:

```
[bits,c,sTX,ref] = TX_BPPM_TH(nBits,Ns,fc,Tc,Ts,Np,Nh,dPPM,Tm,tau,powdBm);
```

2. Introduce asynchronism between users adding a random shift uniformly distributed in $[0, T_s]$ to each interfering user

3. Add noise to the signal obtained as the sum of useful and interfering signals using the `gnoise` function

4. Generate correlation mask and detect with the `corrmask` and `PPMreceiver` functions

Perform and average several runs in order to obtain reliable results

Settings

```
Ni = 5,20,60,200    tau = 0.25e-9   nBits = 1000    powdBm = -30
Ns = 3              Tm = 0.5e-9     dPPM = 0.5e-9   Nh = 6
Ts = 6e-9           fc = 50e9       ebn0 = Inf      Np = 150
Tc = 1e-9
```