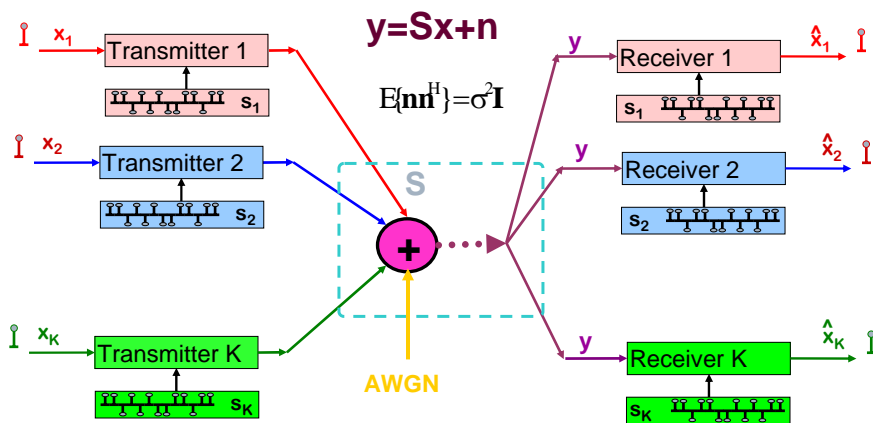


Data Communications Lab 4 (CE00036-3)

- ❖ Simulate a Multiuser Code Division Multiple Access Communication system.
 - ❖ Compare the simulated result with the theoretical expectation.
-

CDMA System Model (K Users)



Multi-user CDMA System Model

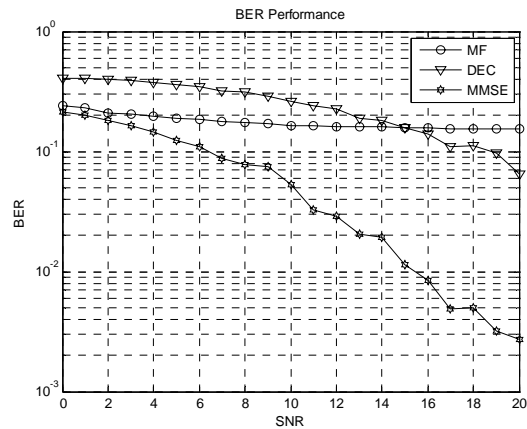
$$\mathbf{y} = \mathbf{S}\mathbf{x} + \mathbf{n}$$

- \mathbf{y} - received signal (N X 1) dimensional vector
 - \mathbf{S} - signature matrix (N X K) dimensional matrix
 - \mathbf{x} - transmitted symbols (K X 1) dimensional vector
 - \mathbf{n} - Gaussian noise (N X 1) dimensional vector
 - N - Number of system **Dimensions (Spreading gain)**
 - K - Number of **Users**
-

Problem Script (1)

- Write a Matlab script to simulate a Multiuser CDMA communication system with BPSK digital Modulation for a signal to noise ratio **0 dB to 13 dB**. Compare the performance of the following Multiuser detectors:
 - Matched Filter (MF)
 - Decorrelator
 - Minimum Mean Squared Error (MMSE)
 - Assume:
 - There are 25 users
 - Signature sequence Length is 31 chips (for gold code), 64 (for Walsh code and random code)
 - Each user transmitting 100 bits per frame.
-

Expected results (random Code)



Gold code generator

- ```

function gold_seq=cyclo(con1, con2)
% MATLAB script for Illustrative Problem 5, Chapter 9.
echo on
% first determine the maximal length shift register sequences
% We'll take the initial shift register content as "00001".
connections1=con1;%[1 0 1 0 0];
connections2=con2;%[1 1 1 0 1];
sequence1=ss_mlsrs(connections1);
sequence2=ss_mlsrs(connections2);
% cyclically shift the second sequence and add it to the first one
L=2^length(connections1)-1;;
for shift_amount=0:L-1,
 temp=[sequence2(shift_amount+1:L)
sequence2(1:shift_amount)];
 gold_seq(shift_amount+1,:)=(sequence1+temp) -
floor((sequence1+temp)./2).*2;
 echo off ;
end;
echo on ;

```

## Gold code generator

---

```
❑ % find the max value of the cross correlation for these sequences
❑ max_cross_corr=0;

❑ for i=1:L-1,
❑ for j=i+1:L,
❑ % equivalent sequences
❑ c1=2*gold_seq(i,:)-1;
❑ c2=2*gold_seq(j,:)-1;
❑ for m=0:L-1,
❑ shifted_c2=[c2(m+1:L) c2(1:m)];
❑ corr=abs(sum(c1.*shifted_c2));
❑ if (corr>max_cross_corr),
❑ max_cross_corr=corr
❑ end;
❑ echo off ;
❑ end;
❑ end;
❑ end;
❑ % note that max_cross_corr turns out to be 9 in this example...
```

---