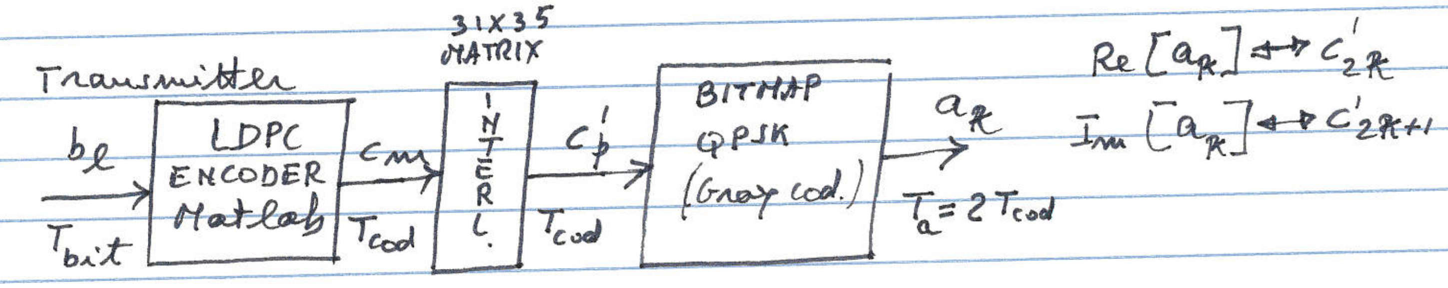
DIGITAL TRANSMISSION

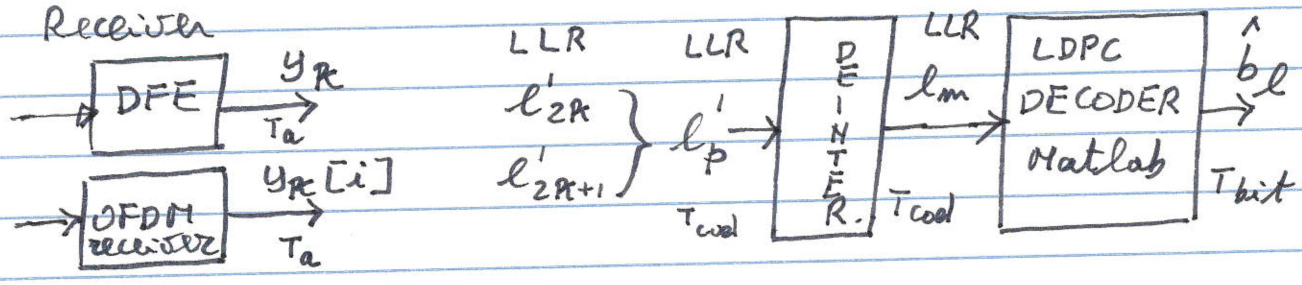
HW4

MAY 26, 2015

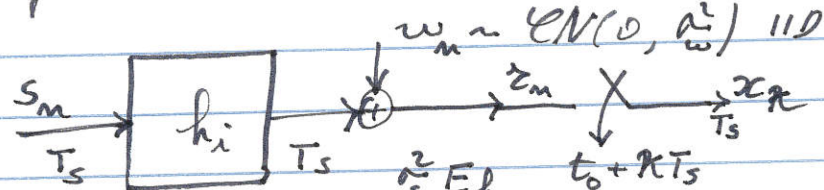
DUE JUNE 8, 2015

Assume that symbols for SC (single carrier) and OFDM systems are generated as follows





With regards to the channel model assume



$$\Gamma = \frac{\sigma_s^2 E_h}{\sigma_w^2}$$

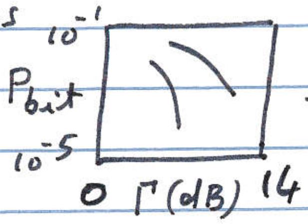
with

$\boxed{1}$ IDEAL CHANNEL : $h_i = \delta_i = \begin{cases} 1, & i = 0 \\ 0, & i \neq 0 \end{cases}$

$\boxed{2}$ DISPERSIVE CH. : $\{h_i\} = \{0, 0, 0, 0, 0, 0.7e^{-j2.57}, 0.24e^{j1.34}, 0.15e^{-j2.66}, 0.58e^{-j1.15}, 0.40e^{-j1.63}, 0, 0, 0\}$

PROBLEM 1 (10 p)

For an ideal ch., plot the following curves obtained by simulations



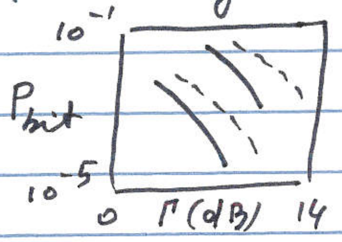
- Uncoded QPSK
- Encoded QPSK with soft input decoder

Write expression of LLR's ℓ'_{2k} and ℓ'_{2k+1}.

PROBLEM 2 (20 p)

For a SC modulator with $T_S = T_a$ and $s_n = a_n$, determine the timing phase t_0 as an integer multiple of T_S.

Plot the following curves



- DFE, uncoded data
- DFE, coded data

} KNOWN CH (solid lines)

- DFE, uncoded data
- DFE, coded data

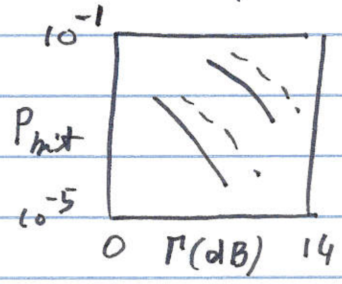
} ESTIMATED CH (dashed lines)

At first assume the ch. is known (both $\{h_i\}$ and σ_w^2).
Design a suitable DFE for each value of Γ. Write values of t_0, M_1, M_2 and D used. Do not change these parameter values as Γ varies. Write expressions of LLR's ℓ'_{2k} and ℓ'_{2k+1}.
Repeat the simulations for an estimated ch. as from HW3 where the training sequence is composed by an M-L sequence of length $L = 21$ binary symbols, partially repeated for a length $N = 7$.
Write expressions of LLR's ℓ'_{2k} and ℓ'_{2k+1}.

PROBLEM 3 (30 p)

For a MC modulator with T_s the sampling period at the ch. input, $M = 512$ and $N_{px} = 7$, determine the timing phase t_0 as the beginning of the cyclic prefix at the receiver (i.e. the DFT starts collecting samples after N_{px} sampling periods).

Plot the following curves

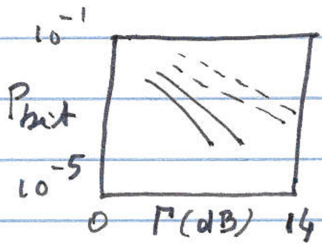


- OFDM, uncoded } KNOWN CH
- OFDM, coded } (solid lines)
- OFDM, uncoded } ESTIMATED CH
- OFDM, coded } (dashed lines)

Write expressions of LLR's $\ell'_{2k}[i]$ and $\ell'_{2k+1}[i]$.

Determine a method to estimate the ch. frequency response and subchannel noise by using a training sequence of only $L = 21$ binary known symbols. Repeat the simulations. Please note: for each Γ where $\sigma_a^2 = 2$, assume that the training sequence is sent at double power, i.e. $\sigma_{a_{TS}}^2 = 4$. Write expressions of LLR's when the estimated ch. is used.

Last. Report on the same figure P_{bit} vs. Γ for SC and OFDM in the presence of coded data and both known ch. and estimated ch.



DFE, coded data } KNOWN CH
OFDM, " " } (solid lines)

DFE, " " } ESTIMATED CH
OFDM, " " } (dashed lines)

```
enc = fec.ldpcenc;  % Construct a default LDPC encoder object

% Construct a companion LDPC decoder object
dec = fec.ldpcdec;
dec.DecisionType = 'Hard decision';
dec.OutputFormat = 'Information part';
dec.NumIterations = 50;
% Stop if all parity-checks are satisfied
dec.DoParityChecks = 'Yes';

% Generate and encode a random binary message
msg = randi([0 1],1,enc.NumInfoBits);
codeword = encode(enc,msg);

% Construct a BPSK modulator object
modObj = modem.pskmod('M',2,'InputType','Bit');

% Modulate the signal (map bit 0 to 1 + 0i, bit 1 to -1 + 0i)
modulatedsig = modulate(modObj, codeword);

% Noise parameters
SNRdB = 1;
sigma = sqrt(10^(-SNRdB/10));

% Transmit signal through AWGN channel
receivedsig = awgn(modulatedsig, SNRdB, 0); ...

% Construct a BPSK demodulator object to compute
% log-likelihood ratios
demodObj = modem.pskdemod(modObj,'DecisionType','LLR', ...
     'NoiseVariance',sigma^2);

% Compute log-likelihood ratios (AWGN channel)
llr = demodulate(demodObj, receivedsig);

% Decode received signal
decodedmsg = decode(dec, llr);

% Actual number of iterations executed
disp(['Number of iterations executed = ' ...
     num2str(dec.ActualNumIterations)]);
% Number of parity-checks violated
disp(['Number of parity-checks violated = ' ...
     num2str(sum(dec.FinalParityChecks))]);
% Compare with original message
disp(['Number of bits incorrectly decoded = ' ...
     num2str(nnz(decodedmsg-msg))]);
```