

# 北京邮电大学实验报告

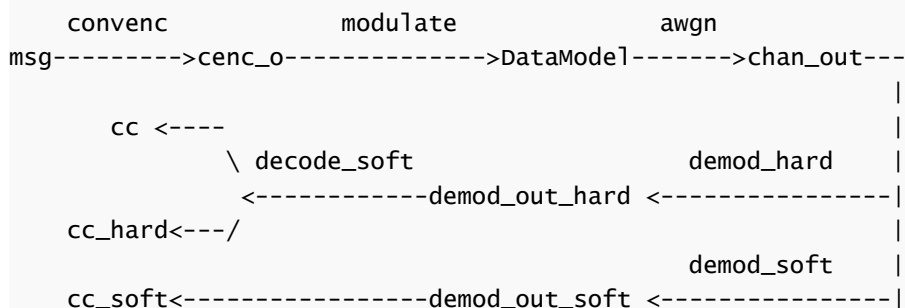
专业：通信工程  
姓名：李昊  
学号：2014210192  
日期：2017-10-09

课程名称：通信系统与仿真 指导老师：赵慧 实验名称：系统传输函数零极点分析

## 一、实验要求

仿真比较硬解调, 硬解调 + 软译码, 软解调 + 软译码

## 二、总体设计



## 三、工程结构

```
.
├── convEncode.m
├── demodulation_hard.m
├── demodulation_soft.m
├── do.m
├── LogMAPdecode.m
├── main_c4_withCoder.m
├── modulation36211.m
└── next.m

0 directories, 8 files
```

## 四、几个小问题

Q1:TREL = poly2trellis( CL, [13, 15, 15, 17]);% in octal %

提问1: 这里说明编码率是多少?

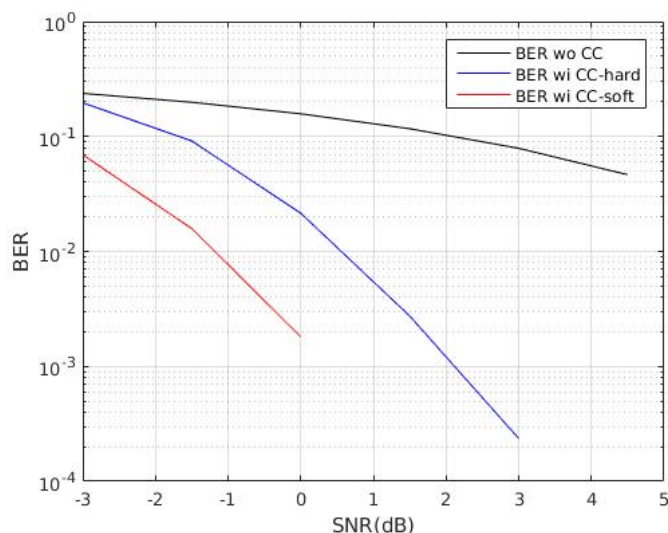
```

A1:R = 1/4
Q2:demod_out_hard = demodulation_hard(chan_out,mod_mode); % 硬解调 %
    提问2: demodulation_hard和demodulation_soft输出的含义有何不同?
A2: demodulation_hard:直接判决,输出的是比特,demodulation_soft:输出的是处理后的复数
Q3: [num_raw, ratio_raw] = biterr(~demod_out_hard, cenc_o); %
    提问3: RawBER是指什么?
A3: 只有硬解调, 没有软译码情况下, 卷积码输出比特的误码率.
Q4:decoder_in_soft = Inf*(2*demod_out_hard-1); %
    提问4: 这句代码的作用是什么? 为什么要有这句
A4: 硬解调的输出是[ 0 1], 要送入软译码器, 须模拟极化.[0->-inf, 1->inf]

```

## 五、 结果分析

### (1) 仿真



(2) 相对于硬解调, 加上软译码效果会提高, 如果是软解调加上软译码效果会大幅增加

(3) 在硬解调的情况下, 加上软译码, 在同等的 BER 下, 可减少 1.5dB 的 SNR

## 六、 具体实现

```

%% 此代码是用来演示 软解调 的主函数
%% 包含 卷积编码, AWGN
%% 请运行该代码 (时间比较长), 给出结果, 说明结果要说明什么?
clear all
close all
clc
%%***** simulation parameter *****

NumBit = 100; % Mesg Length excluding pre-determined bits for starting
               & ending trellis at state 0

```

```

mod_mode=2;
SNR_arr =[-3:1.5:5];
num_snr=length(SNR_arr);
Nframe = 80*(1:num_snr);
ber_hard=zeros(1,num_snr);
ber_soft=zeros(1,num_snr);
ber_raw=zeros(1,num_snr);
%% Convolutional Code 参数
CL = 4 ;% constraint length
TREL = poly2trellis( CL, [13, 15, 15, 17]);% in octal %
    提问1: 这里说明编码率是多少?
%-----R=1/4-----
NumBit = NumBit + 2*(CL-1); % space for start & tail bits
Inf=10^6; %表示特别大的无穷
%%***** 循环 *****
for loop=1:num_snr
    SNR=SNR_arr(loop);
    %disp([' SNR: ' num2str(SNR)]);
    No = 10^(-SNR/10);
    signal_power = 1;
    Es = signal_power;%log2(TREL.numOutputSymbols)*signal_power;
    Lc = 4*Es/No; % 软译码所用参数
    for num_frame = 1:Nframe(loop)
%%***** 发端 *****
        %% Rate 1/4 encoder with 8-states used in 3GPP cellular 3G/4G standard
        msg(1 : CL-1) = zeros(1,CL-1); % 1st (CL-1) bits must be 0
        msg(CL : NumBit -CL+1) = randi([0 1],1,NumBit -2*CL +2) ; % Random data
        msg(NumBit -CL + 2 : NumBit) = zeros(CL-1, 1); % Last (CL-1) bits must
        be 0
        %disp(['length of msg:' num2str(length(msg))])

        [cenc_o, final_state] = convenc(msg, TREL);
        %disp(['length of cenc_o:' num2str(length(cenc_o))])

        %% modulation

        DataModed=modulation36211(cenc_o,mod_mode);
        %disp(['length of DataModed:' num2str(length(DataModed))])

        %% Generate and add Gaussian-noise mean=0, variance= noise_power
        chan_out = awgn(DataModed, SNR);
        %disp(['length of chan_out:' num2str(length(chan_out))])

%%***** 接收端 *****
        %% demodulation soft or hard
        demod_out_hard = demodulation_hard(chan_out,mod_mode); % 硬解调 %
        提问2: demodulation_hard和demodulation_soft输出的含义有何不同?
        %demodulation_hard:直接判决,输出的是比特
        %disp(['length of demod_out_hard:' num2str(length(demod_out_hard))])
    end
end

```

```
    demod_out_soft = demodulation_soft(chan_out,mod_mode,10^(SNR/10)); %
软解调
%demodulation_soft:输出的是处理后的复数
%disp(['length of demod_out_soft:' num2str(length(demod_out_soft))])

    %% 解调出来直接对比bit error= RawBER
    [num_raw, ratio_raw] = biterr(~demod_out_hard, cenc_o); %
提问3: RawBER是指什么?
%只有硬解调, 没有软译码情况下, 卷积码输出比特的误码率.
    ber_raw(loop) = ber_raw(loop) + num_raw;
    %% 硬解调信息 送入软译码器 软译码器要求输入是LLR
    decoder_in_soft = Inf*(2*demod_out_hard-1); %
提问4: 这句代码的作用是什么? 为什么要有这句
% 硬解调的输出是[ 0 1], 要送入软译码器, 须模拟极化.[0->-inf, 1->inf]
    [decd_msg_hard,LLR_hard, Alpha_hard, Beta_hard] = LogMAPdecode(TREL,
decoder_in_soft, Lc);
    [num_hard, ratio_hard] = biterr(decd_msg_hard, msg);
    ber_hard(loop) = ber_hard(loop) + num_hard;
    %% 软解调信息 送入软译码器
    [decd_msg_soft,LLR_soft, Alpha_soft, Beta_soft] = LogMAPdecode(TREL,
demod_out_soft, Lc);

    [num_soft, ratio_soft] = biterr(decd_msg_soft, msg);
    ber_soft(loop) = ber_soft(loop) + num_soft;
    %break;
end
% break;
end
SNR;
ber_raw = ber_raw./Nframe./(NumBit*4);
ber_hard = ber_hard./Nframe./NumBit;
ber_soft = ber_soft./Nframe./NumBit;
semilogy(SNR_arr,ber_raw,'k-');hold on;
semilogy(SNR_arr,ber_hard,'b-');hold on;
semilogy(SNR_arr,ber_soft,'r-');
grid on;xlabel('SNR(dB)');ylabel('BER');
legend('BER wo CC','BER wi CC-hard','BER wi CC-soft');
ylim([0.0001,1])
grid on;
```