



**北京航空航天大学**  
B E I H A N G U N I V E R S I T Y

**通信原理**  
**MATLAB 软件作业三**

**AWGN 信道中 BPSK 调制解调**  
**及误码率性能分析**

学 院 电子信息工程学院  
作 者 何沃洲  
学 号 13021264

**2016 年 6 月**

# 目 录

（一）实验原理.....	1
1.1 二进制数字相位调制（BPSK）的原理.....	1
1.2 BPSK 的相干解调原理.....	2
（二）MATLAB 程序及仿真结果 .....	3
2.1 变量初始化.....	3
2.2 基带信号成形及 BPSK 调制.....	4
2.3 BPSK 信号经过 AWGN 信道后解调 .....	5
2.4 抽样判决.....	7
2.5 BPSK 误码率与 AWGN 信道信噪比的关系曲线 .....	8
（三）结论与收获.....	11
附录.....	12

## （一）实验原理

### 1.1 二进制数字相位调制（BPSK）的原理

二进制相移键控（BPSK）是利用载波的初相位直接表示数字信号的调制方式。BPSK 中载波的相位随调制信号 1 或 0 而改变，通常用相位 0 和  $\pi$  来分别表示 1 或 0。二进制相移键控已调信号的表达式为：

$$S_{BPSK}(t) = \left[ \sum_n a_n g(t - nT) \right] \cos(\omega t)$$

其中， $g(t)$  是宽度为  $T_b$  的门函数， $a_n = \begin{cases} +1, & \text{概率 } P \\ -1, & \text{概率 } (1 - P) \end{cases}$

$a_n$  为基带信号码元， $\omega$  为载波角频率。在形式上，BPSK 与 ASK 类似，也可以看成是双极性非归零码经过双边带调制的产物，因此也属于线性调制。BPSK 调制信号无直流分量，相当于模拟调制中的抑制载波双边带调制。

BPSK 的功率谱与 OOK 信号相同，只是少了一个离散的载频分量。这一结论同样适用于基带信号为其他形式时的 BPSK 信号。

BPSK 调制器可以采用相乘器，也可以用相位选择器来实现。

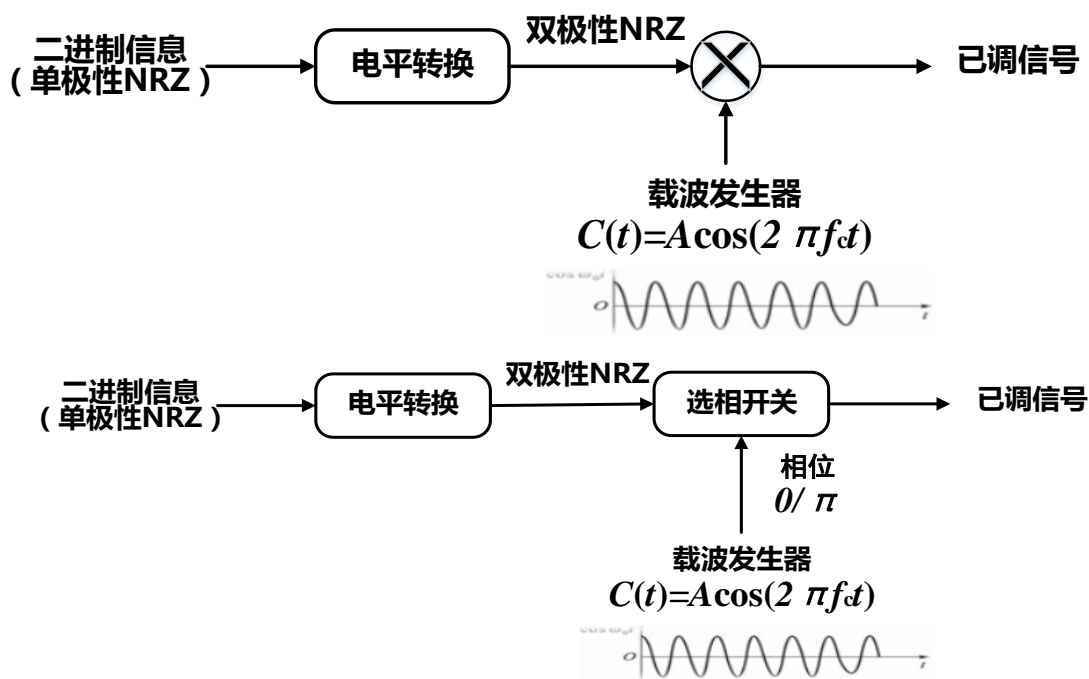


图 1 BPSK 调制器：相乘法（上）；相位选择法（下）

## 1.2 BPSK 的相干解调原理

BPSK 解调必须采用相干解调的方式,如何得到同频同相的载波是个关键问题。由于 BPSK 信号不存在载频分量,因而无法从已调信号直接用滤波法提取本地载波。常用的载波恢复电路是平方环和科斯塔斯环。

在接收端用一个同频本振与接收信号相乘再通过低通滤波器进行抽样判决,同相时得到正值,反相时得到负值。在这样的解调方式下,由于接收端使用的本振可能与理想本地载波同相也可能反相,即存在  $0$ 、 $\pi$  相位模糊度,所以导致了 BPSK 信号的解调难以避免译码后得到的码元可能与发送端的码元极性相反。

BPSK 这种使用载波信号相位直接变化来传递信息的方式称为绝对调相,为了解决绝对调相解调时的相位模糊问题,于是有了相对调相方式,以载波的相对相位变化作为信息载体,避免了相位模糊的问题。

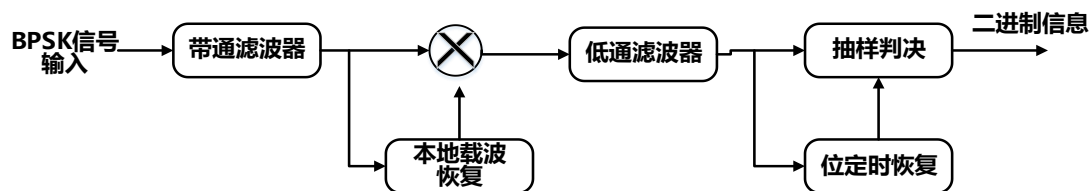


图 2 BPSK 相干解调原理

此外, BPSK 信号还可以利用最佳接收机进行解调和判决。根据最大似然准则设计的最佳接收机如下面图 3 所示:

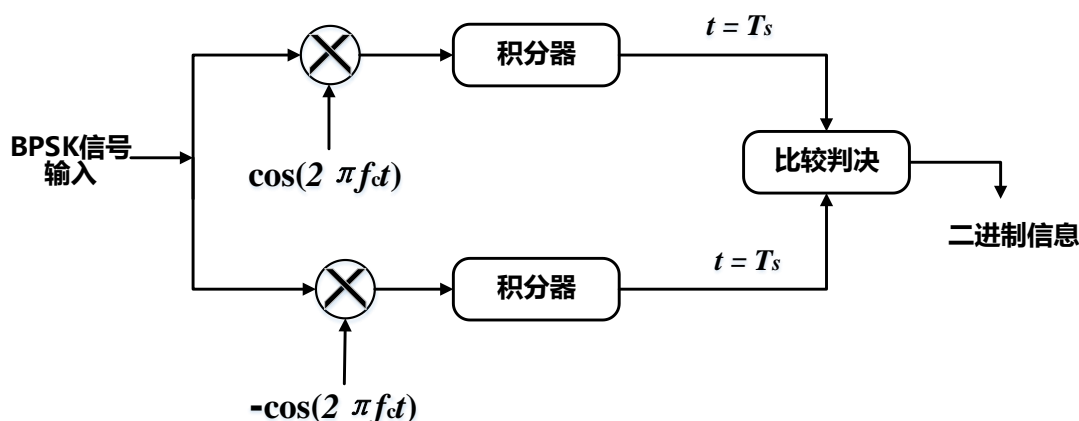


图 3 BPSK 最佳接收机

## (二) MATLAB 程序及仿真结果

### 2.1 变量初始化

```
clear;
SNR = 10;    %信道信噪比 (dB)
num = 10e3;  %码元个数
len = 20;    %码元长度
fc = 1000;   %载波频率 (Hz)
fs = len*fc; %采样频率 (Hz)
A = 1;       %载波幅度
ts=1/fs;     %采样的时间步长
t0 = 10;     %仿真时长 (秒)
t=ts:ts:t0;  %时间轴序列
ct = A*sin(2*pi*fc*t); %载波
bt = [];     %基带信号
rebt = [];   %恢复后的基带信号
n = 20;      %绘图画出的周期个数
N = 2^18;    %FFT分析点数
code = randi([0,1],1,num); %产生等概分布的基带信号序列
decode = zeros(1,num);    %判决后的基带信号序列
```

#### 【注解】

表 1 变量声明

变量名称	变量符号	变量值
信噪比	$SNR$	10dB
载波频率	$f_c$	1kHz
采样频率	$f_s$	20kHz
码长	$num$	10000
码长 (采样点)	$len$	20
仿真时长	$t_0$	10s
载波	$ct$	/
基带信号/复原信号	$bt / rebt$	/
已调信号	$st$	/
基带/复原信号波形	$code / decode$	/

Main 函数的第一部分首先进行了仿真参数的初始化，共采集 200,000 个数据点。采样频率满足奈奎斯特抽样定理。

通过 MATLAB 随机函数 randi 生成要发送的 0、1 数据序列。

## 2.2 基带信号成形及 BPSK 调制

```

for i = 1:1:num %基带信号成形
    if (code(i) ==0)
        cp = zeros(1,len);
    else
        cp = ones(1,len);
    end
    bt = horzcat(bt,cp);
end
bt2 = 2*bt-1; %对应的双极性基带信号
st = ct.*bt2; %BPSK调制后的信号

%-----
%作图
%-----
figure('NumberTitle','off','Name','BPSK调制');
subplot(2,1,1);
plot(t(1:n*len),bt(1:n*len),'-b'); %作图：基带信号
hold on
grid
axis([0,n*len/fs,-0.1,1.1])
xlabel('\itt','FontName','Times New Roman','FontSize',12);
ylabel('\itb(t)','FontName','Times New Roman','FontSize',12);
title('基带信号','FontName','Times New Roman','FontSize',12);

subplot(2,1,2);
plot(t(1:n*len),st(1:n*len),'-b'); %作图：BPSK调制信号
axis([0,n*len/fs,-1,1])
hold on
grid
xlabel('\itt','FontName','Times New Roman','FontSize',12);
ylabel('\its(t)','FontName','Times New Roman','FontSize',12);
title('BPSK调制信号','FontName','Times New Roman','FontSize',12);

```

### 【注解】

通过逐个扫描数据序列进行高低电平的拼接形成基带信号波形。BPSK 调制选用相乘法实现：首先把波形转换为双极性。再与载波相乘。得到的波形如下面图 4 所示：

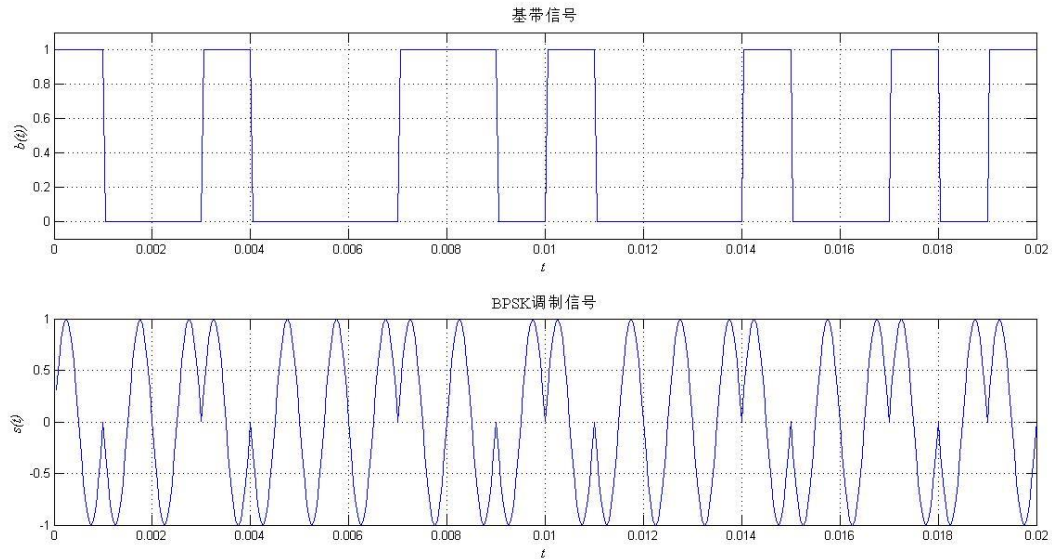


图 4 基带信号波形及 BPSK 调制后波形

## 2.3 BPSK 信号经过 AWGN 信道后解调

```

%-----
% BPSK信号经过AWGN信道后相干解调
%-----
st_channel = add_noise(st,SNR); %加性高斯白噪声
st_channel = filterdesign(ceil(2*fc*N/fs),N,len*num,st_channel); %接收信号先经过带通滤波器滤除带外噪声
dmst1 = st_channel.*ct; %乘以本地的相干载波
dmst = filterdesign(ceil(fc*N/fs),N,len*num,dmst1); %经过低通滤波器
%作图
figure('NumberTitle','off','Name',' BPSK信号经过AWGN信道后相干解调');
subplot(4,1,1); %作图: BPSK调制信号
plot(t(1:n*len),st(1:n*len),'-b');
axis([0,n*len/fs,-1,1])
hold on
grid
xlabel('\itt','FontName','Times New Roman','FontSize',12);
ylabel('\its(t)','FontName','Times New Roman','FontSize',12);
title(' BPSK调制信号','FontName','Times New Roman','FontSize',12);

subplot(4,1,2); %作图: BPSK调制信号经过AWGN信道
plot(t(1:n*len),st_channel(1:n*len),'-b');
axis([0,n*len/fs,-1,1])
hold on
grid
xlabel('\itt','FontName','Times New Roman','FontSize',12);
ylabel('\its(t) channel','FontName','Times New Roman','FontSize',12);
title('通过AWGN信道后的BPSK信号 (SNR=10dB)','FontName','Times New Roman','FontSize',12);

```

```

subplot(4,1,3);      %作图: BPSK调制信号乘以相干载波
plot(t(1:n*len),dmst1(1:n*len),'-b');
axis([0,n*len/fs,-1,1])
hold on
grid
xlabel('\itt','FontName','Times New Roman','FontSize',12);
ylabel('\itdm s(t) ','FontName','Times New Roman','FontSize',12);
title(' BPSK解调信号 (未滤波)','FontName','Times New
Roman','FontSize',12);

subplot(4,1,4);      %作图: BPSK调制信号乘以相干载波后通过低通滤波器
plot(t(1:n*len),dmst(1:n*len),'-b');
axis([0,n*len/fs,-1,1])
hold on
grid
xlabel('\itt','FontName','Times New Roman','FontSize',12);
ylabel('\itdm s(t) ','FontName','Times New Roman','FontSize',12);
title(' BPSK解调信号','FontName','Times New Roman','FontSize',12);

```

### 【注解】

BPSK 信号在 AWGN 中叠加特定信噪比下的高斯白噪声。进入接收端首先通过带通滤波器滤除带外噪声，然后与本地的理想载波相乘，最后通过低通滤波器得到解调波形（其中用到的滤波器和加噪声子函数可见附录）。这个过程中各点的波形如下面图 5 所示：

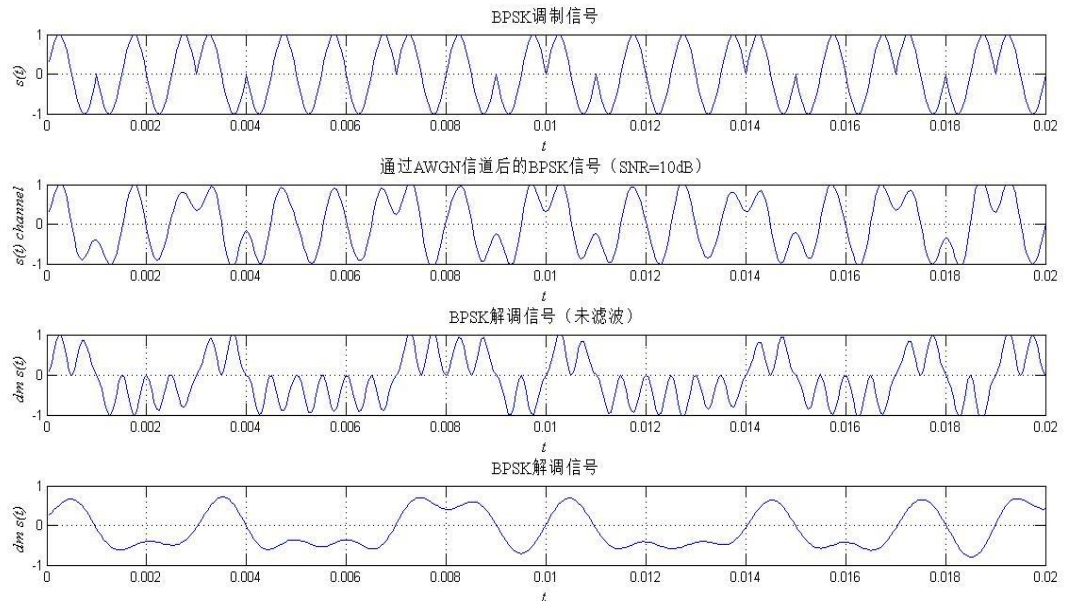


图 5 BPSK 信号经过 AWGN 信道后解调过程中的各点波形



## 2.4 抽样判决

```

%-----
%抽样判决
%-----
for k=1:num
    if (dmst(-len/2+len*k)>0) %取码元周期的中点进行抽样判决
        decode(k) = 1;
    else
        decode(k) = 0;
    end
end
for j=1:num %恢复基带信号成形
    if (decode(j) == 0)
        cp2 = zeros(1,len) ;
    else
        cp2 = ones(1,len);
    end
    rebt = horzcat(rebt,cp2);
end

figure('NumberTitle', 'off', 'Name', '发送与接收端的基带信号对比');
subplot(2,1,1);
plot(t(1:n*len),bt(1:n*len),'-b'); %作图：基带信号
hold on
grid
axis([0,n*len/fs,-0.1,1.1])
xlabel('\itt','FontName','Times New Roman','FontSize',12);
ylabel('\itb(t)','FontName','Times New Roman','FontSize',12);
title('基带信号','FontName','Times New Roman','FontSize',12);

subplot(2,1,2);
plot(t(1:n*len),rebt(1:n*len),'-b'); %作图：恢复后的基带信号
hold on
grid
axis([0,n*len/fs,-0.1,1.1])
xlabel('\itt','FontName','Times New Roman','FontSize',12);
ylabel('\itrebt(t)','FontName','Times New Roman','FontSize',12);
title('复原的基带信号','FontName','Times New Roman','FontSize',12);

```

### 【注解】

解调后与本地载波同相则解调波形采样点为正，反相则为负。选择码元周期的中点进行以零为标准的抽样判决，得到发送的数据序列，再进行基带波形成形。与发送端的基带波形对比如下面图 6 所示：

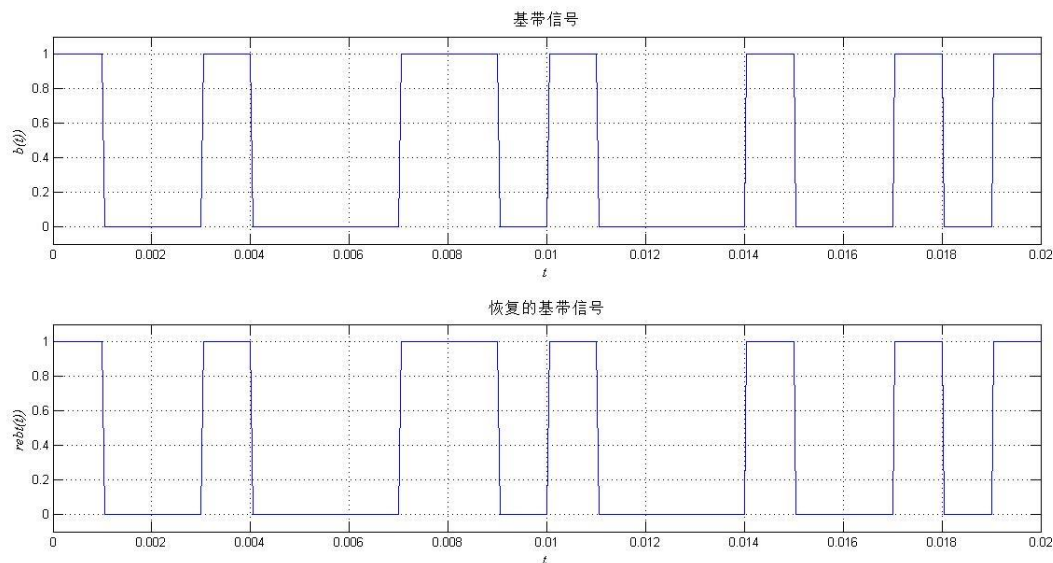


图 6 原始基带信号和恢复后的基带信号对比

## 2.5 BPSK 误码率与 AWGN 信道信噪比的关系曲线

```
%-----
%绘制误码率与信噪比的关系曲线
%-----
figure('NumberTitle', 'off', 'Name', '误码率与信噪比的关系曲线');
spot = 60; %绘图的描点数(+1)
range = 30; %仿真的信噪比范围
error_rate = zeros(1,spot);
SNR_array = zeros(1,spot);
for m = 1:1:spot % SNR步进
    SNR_array(m) = -30+range/spot*m;
    error_rate(m) = error_SNR(SNR_array(m),code,st);
    hold on
end

plot(SNR_array,error_rate,'*k',SNR_array,qfunc(2*sqrt(10.^(SNR_array/10))),'-b'); %在误码率与SNR关系曲线上逐个描点

grid
legend('仿真曲线','理论曲线');
set(gca,'YScale','log') %纵坐标采用对数坐标
xlabel('信噪比(dB)','FontName','Times New Roman','FontSize',12);
ylabel('误码率','FontName','Times New Roman','FontSize',12);
title('误码率与信噪比之间的的关系曲线');

%-----
%最佳接收条件下误码率与信噪比的关系曲线
%-----
figure('NumberTitle', 'off', 'Name', '最佳接收条件下误码率与信噪比的关系曲线
```

```

');
spot = 60; %绘图的描点数(+1)
range = 30; %仿真的信噪比范围
error_rate2 = zeros(1,spot);
SNR_array = zeros(1,spot);
for m = 1:1:spot % SNR步进
    SNR_array(m) = -30+range/spot*m;
    error_rate2(m) = error_SNR_best_receiver(SNR_array(m),code,st);
    hold on
end

plot(SNR_array,error_rate,'*k', SNR_array,error_rate,'*g');
%在误码率与SNR关系曲线上逐个描点

grid
legend('仿真曲线','最佳接收');
set(gca,'YScale','log') %纵坐标采用对数坐标
xlabel('信噪比(dB)','FontName','Times New Roman','FontSize',12);
ylabel('误码率','FontName','Times New Roman','FontSize',12);
title('最佳接收条件下误码率与信噪比之间的的关系曲线');

```

### 【注解】

通过 SNR 从-30dB 到 0dB 步长 0.5dB 的扫描，逐个统计出特定 SNR 下的误码率，绘出关系曲线。把理论曲线也绘于图中便于对比（误码率的纵坐标采用对数刻度与教材一致）。为了便于模块化的调用，把以上 BPSK 调制解调的过程集成到 error\_SNR 函数中。

此外，为了探究最佳接收条件对误码率的影响，实验还进行了最大似然准则最佳接收机和普通相干解调在各个信噪比下误码率的对比。

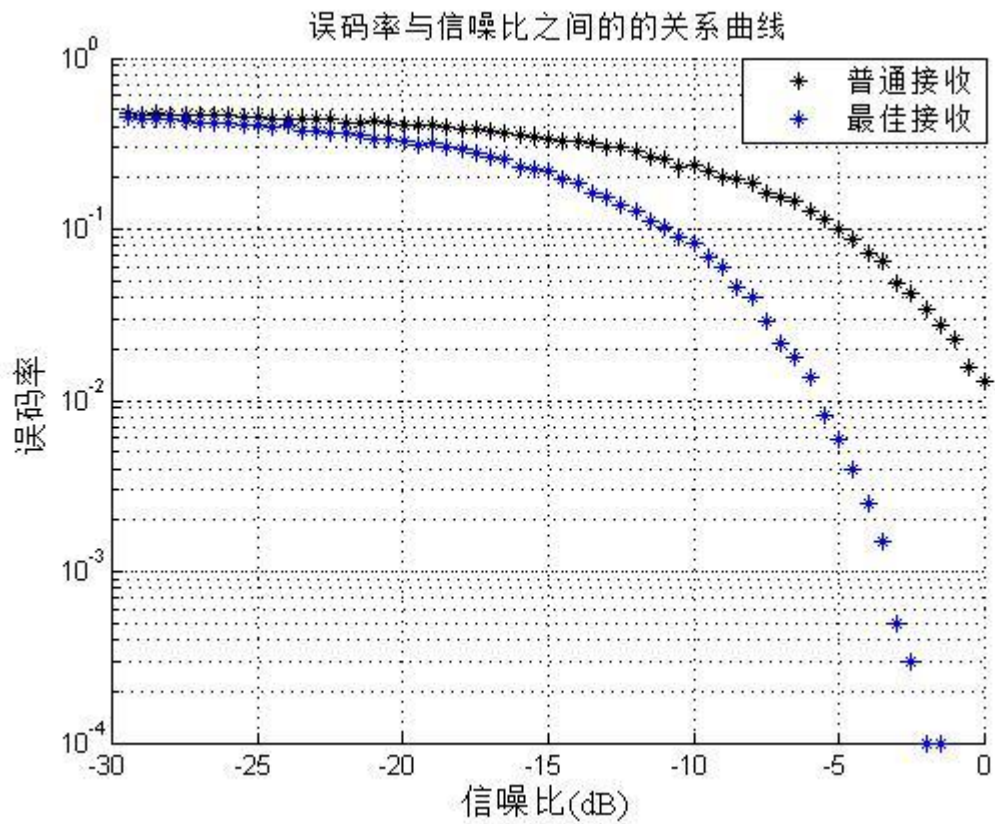
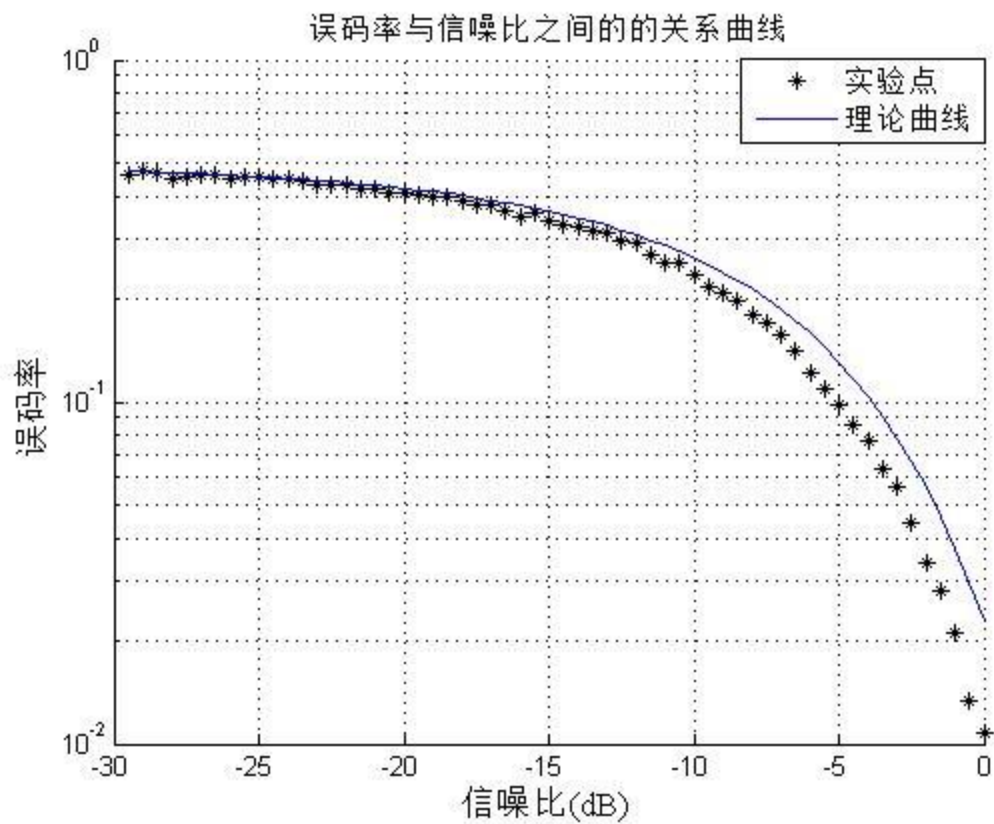


图 7 BPSK 信号误码率与信噪比的关系曲线（上）

最佳接收和普通相干解调的误码率对比（下）

### （三）结论与收获

经过对于这一系列关于 BPSK 调制解调及误码率性能的 MATLAB 仿真，我对它的原理和过程有了更深入的理解和认识。

BPSK 调制解调的波形是在较高信噪比的条件下（ $\text{SNR}=10\text{dB}$ ）仿真的，由于每次发送的码长只有 10000，误码率分辨力  $10^{-4}$ ，所以该条件下接收的数据序列基本与发送序列一致（该信噪比下理论的误码率约为  $10^{-5}\sim 10^{-6}$  数量级）。

绘制误码率与 SNR 关系曲线选择的是低信噪比的一段（ $-30\text{dB}\sim 0\text{dB}$ ），在该段中可以观察到误码率随信噪比的明显变化，从  $-30\text{dB}$  时 50% 的误码率（可以视为随机信号，失去信息传输功能），逐渐变化到  $10^{-2}$  以下。由于码长决定的误码率精度有限，所以信噪比较高的一段与理论曲线略有偏差，但总体来说还是基本吻合的，较成功地复现了 BPSK 误码率性能随信噪比的变化关系。

在最佳接收机与普通相干解调的对比中，可以明显看出最佳接收在同样的信噪比条件下误码率有较大改善，与教材的理论推导也是吻合的。

## 附录

Main 函数中用到的几个函数：

### filterdesign.m

```
function [ output ] = filterdesign(N_filter,N_FFT,data_length,input)
%在频域上设计理想滤波器
filter1 = ones(1,N_filter);
filter2 = zeros(1,N_FFT/2 - N_filter);
filter0 = [filter1,filter2,filter2,filter1];
%输入信号的频谱
input_spectrum = fft(input,N_FFT);
%输入信号经过滤波器后的频谱
output_spectrum = input_spectrum.*filter0;
%输入信号经过滤波器后的时域采样
output_spectrum_ifft = real(ifft(output_spectrum,N_FFT));
output = real(output_spectrum_ifft(1:round(data_length)));
end
```

### error\_counter.m

```
function [ error_rate] = error_counter( code,decode,num )

error = abs(code-decode); %误码序列
error_num = 0;
for j = 1:1:num
    if error(j)~=0
        error_num = error_num+1; %对误码计数
    end
end

error_rate = error_num/num; %计算误码率
end
```

### add\_noise.m

```
function [ Y,noise ] = add_noise( X,SNR ) function [ Y,noise ] =
add_noise( X,SNR ) %AWGN信道加性高斯白噪声
noise = randn(size(X));
noise = noise - mean(noise);
signal_power = 1/length(X)*sum(X.^2);
noise_variance = signal_power/(10^(SNR/10));
noise = sqrt(noise_variance)/std(noise)*noise;
Y = X + noise;
end

noise = randn(size(X));
noise = noise - mean(noise);
signal_power = 1/length(X)*sum(X.^2);
noise_variance = signal_power/(10^(SNR/10));
noise = sqrt(noise_variance)/std(noise)*noise;
```

```
Y = X + noise;
end
```

### error\_SNR.m

```
function [error_rate] = error_SNR( SNR )

SNR = 10;    %信道信噪比 (dB)
num = 10e3;  %码元个数
len = 20;    %码元长度
fc = 1000;   %载波频率 (Hz)
fs = len*fc; %采样频率 (Hz)
A = 1;       %载波幅度
ts=1/fs;     %采样的时间步长
t0 = 10;     %仿真时长(秒)
t=ts:ts:t0;  %时间轴序列
ct = A*sin(2*pi*fc*t); %载波
N = 2^18;    %FFT分析点数
decode = zeros(1,num);    %判决后的基带信号序列

st_channel = add_noise(st,SNR); %加性高斯白噪声
passband = ceil(2*fc*N/fs);    %数字滤波器的等效截止频率
st_channel = filterdesign(passband,N,len*num,st_channel);
%接收信号先经过带通滤波器滤除带外噪声
dmst1 = st_channel.*ct; %乘以本地的相干载波
dmst = filterdesign(passband,N,len*num,dmst1); %经过低通滤波器

for k=1:num
    if(dmst(-len/2+len*k)>0) %取码元周期的中点进行抽样判决
        decode(k) = 1;
    else
        decode(k) = 0;
    end
end

error_rate = error_counter(code,decode,num); %计算该信噪比下的误码率
```

### error\_SNR\_best\_receiver.m

```
function [error_rate] = error_SNR_best_receiver( SNR )

SNR = 10;    %信道信噪比 (dB)
num = 10e3;  %码元个数
len = 20;    %码元长度
fc = 1000;   %载波频率 (Hz)
fs = len*fc; %采样频率 (Hz)
A = 1;       %载波幅度
ts=1/fs;     %采样的时间步长
t0 = 10;     %仿真时长(秒)
t=ts:ts:t0;  %时间轴序列
```

```
ct = A*sin(2*pi*fc*t); %载波
N = 2^18; %FFT分析点数
code = randi([0,1],1,num); %产生等概分布的基带信号序列
decode = zeros(1,num); %判决后的基带信号序列

st_channel = add_noise(st,SNR); %加性高斯白噪声
dmst1 = st_channel.*ct; %乘以本地的相干载波s1(t)
dmst2 = st_channel.*(-ct); %乘以本地的相干载波s2(t)

sum1 = zeros(1,num); %积分变量1
sum2 = zeros(1,num); %积分变量2
for k = 1:1:num
    for xk = 1:1:len %比较判决得到数据序列
        sum1(k) = sum1(k)+dmst1(len*(k-1) + xk);
        sum2(k) = sum2(k)+dmst2(len*(k-1) + xk);
    end

    if sum1(k)>sum2(k) %恢复数据序列
        decode(k) = 1;
    else
        decode(k) = 0;
    end
end

error_rate = error_counter(code,decode,num); %计算该信噪比下的误码率

end
```