

密 级	
编 号	



浙江大學

Zhejiang University

通信原理

QPSK 数字通带传输仿真

QPSK Digital Passband Transmission Simulation

学 院	信息与工程学院
专 业	信息工程
姓 名	
学 号	

2023 年 6 月 7 日

目 录

第 1 章 题目要求	1
1.1 题目	1
1.2 要求	1
1.3 分析	1
第 2 章 QPSK 数字通带传输原理	2
2.1 QPSK 调制	2
2.2 QPSK 解调	3
第 3 章 仿真过程	4
第 4 章 仿真结果	8
4.1 10dB 高斯信道传输结果	8
4.2 误码率统计结果	9
4.3 不同信噪比接收图像	10

第 1 章 题目要求

1.1 题目

请构建基于 QPSK 的数字通带传输系统，考虑加性高斯白噪声信道条件下，用 MATLAB/C(C++) 仿真采用相干解调时的数字通带传输系统。

1.2 要求

1. 给出误码率的统计结果，与理论结果进行比较；
2. 发送下述图像，画出接收信噪比为 0dB, 5dB, 10dB, 15dB 的接收图像。



图 1-1 发送图像

1.3 分析

用 MATLAB 仿真传输该 RGB 图像，首先需要得到图像的数据张量，之后将张量转为一维向量，再统计各个值的频率，采用霍夫曼编码对图像数据进行编码，得到二进制比特流，再进行传输。

但该方法存在的问题是，由于霍夫曼编码是不等长编码，所以当传输存在误码时，再进行霍夫曼解码所得到的数据个数与原始数据可能不相等，所以解码得到的数据无法恢复为原始大小的张量，进而也无法用图像进行展示。所以该方法在误码传输时不适用。

若想传输 RGB 图像，则只能采取等长编码的方法，这意味着每个数据需要 256 个比特进行编码，有极大的数据冗余，且这样传输出 RGB 意义不大，并且由于数据量过大，对计算机性能要求也比较高。

综上所述，在本实验中，我们采取将 RGB 图像转化为灰度图，并且因为该图像颜色数目少，我们再将其转化为二值图，即二进制比特流直接代表图像，由此进行仿真。

第 2 章 QPSK 数字通带传输原理

2.1 QPSK 调制

一般情况下 MPSK 调制信号再时间间隔 $kT \leq t \leq (k+1)T$ 中的相位为 ϕ_k , ϕ_k 可以取 M 个不同的取值, 所以:

$$\begin{aligned} s(t) &= \sum_{k=-\infty}^{\infty} \cos(2\pi f_c t + \theta + \phi_k) g_T(t - kT) \\ &= x(t) \cos(2\pi f_c t + \theta) - y(t) \sin(2\pi f_c t + \theta) \end{aligned} \quad (2-1)$$

其中

$$x(t) = \sum_{k=-\infty}^{\infty} I_k g_T(t - kT), \quad y(t) = \sum_{k=-\infty}^{\infty} Q_k g_T(t - kT) \quad (2-2)$$

$$I_k = \cos \phi_k, \quad Q_k = \sin \phi_k \quad (2-3)$$

$$g_T(t) = \begin{cases} A & 0 \leq t < T \\ 0 & \text{others} \end{cases} \quad (2-4)$$

在 M 元 PSK 信号中相移 ϕ_k 可以取 M 个值, 一般 $\phi_k = \pi(2a_k + N)/M$, $a_k = 0, 1, \dots, M-1$, 其中 $N = 0$ 或 $N = 1$ 。对于四元 PSK 调制信号, 即 QPSK, 相位角对应取值如图 2-1。

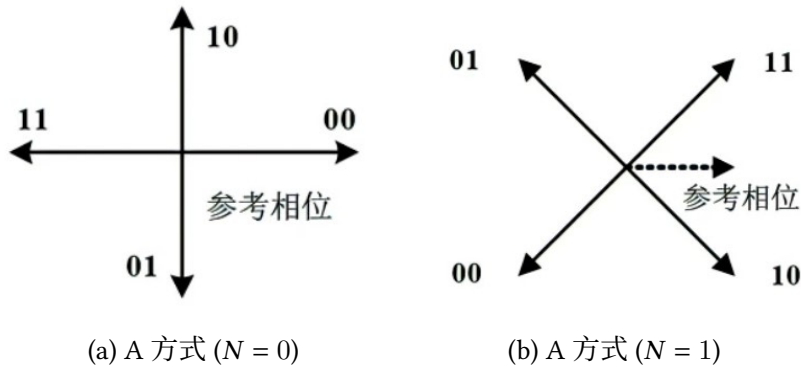


图 2-1 QPSK 相位角的两种取值

QPSK 调制原理框图如图 2-2 所示. 它是对两路正交载波进行对称幅度调制来实现的。相应的编码映射如表 2-1 所示。

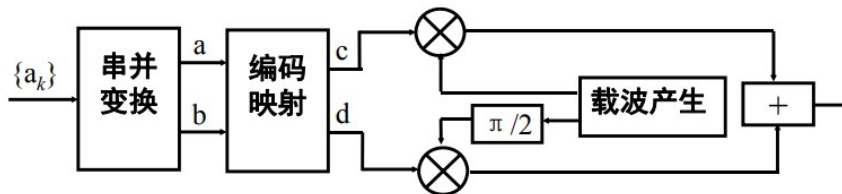


图 2-2 QPSK 调制器原理框图

表 2-1 QPSK 编码映射

a	b	N = 0		N = 1	
		(c,d)	φ_k	(c,d)	φ_k
0	0	(1,0)	0	$(1,1)/\sqrt{2}$	$\pi/4$
1	0	(0,1)	$\pi/2$	$(-1,1)/\sqrt{2}$	$3\pi/4$
1	1	(-1,0)	π	$(-1,-1)/\sqrt{2}$	$5\pi/4$
0	1	(0,-1)	$3\pi/2$	$(1,-1)/\sqrt{2}$	$7\pi/4$

2.2 QPSK 解调

对于 $M = 4$, $N = 1$ 的 QPSK, 可以写成

$$\begin{aligned}
 s(t) &= \sum_{k=-\infty}^{\infty} \cos(2\pi f_c t + \varphi_k) g_T(t - kT) \\
 &= \sum_{k=-\infty}^{\infty} \{ \cos \varphi_k \cdot g_T(t - kT) \cdot \cos[2\pi f_c(t - kT)] - \\
 &\quad \sin \varphi_k \cdot g_T(t - kT) \cdot \sin[2\pi f_c(t - kT)] \}
 \end{aligned} \tag{2-5}$$

其中 $\varphi_k \in \left\{ \frac{2m+1}{4}\pi, \quad m = 0, 1, 2, 3 \right\}$, 在一个符号间隔 $(0, T)$, 码元信号为:

$$\begin{aligned}
 s_m(t) &= \cos \varphi_m g_T(t) \cos(2\pi f_c t) - \sin \varphi_m g_T(t) \sin(2\pi f_c t) \\
 &= \frac{\sqrt{2}}{2} (I \cos(2\pi f_c t) - Q \sin 2\pi f_c t) g_T(t), \quad m = 0, 1, 2, 3
 \end{aligned} \tag{2-6}$$

其中 $I = \pm 1$, $Q = \pm 1$ 。

QPSK 信号相当于两路正交载波的 BPSK, 它的相干解调原理框图如图 5-3 所示

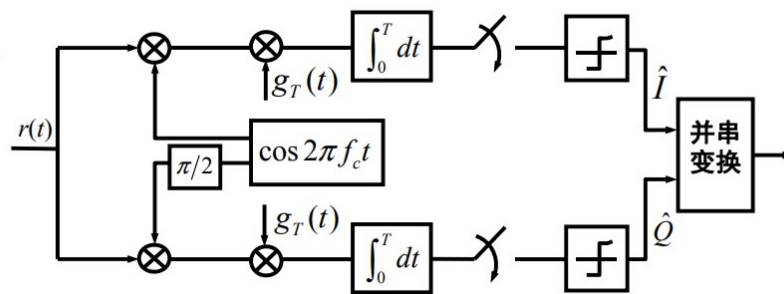


图 2-3 QPSK 信号相干解调原理框图

第3章 仿真过程

本实验仿真环境采用 MATLAB。首先获取信源信息。在本实验中，所传递的信息是图片1-1，由于原图像是 RGB 图像，要实现 RGB 到二进制码的相互转化比较复杂，为了方便传输，我们首先将图像转化为灰度图，再计算其灰度阈值，进而将图像转为二值图像，得到用于传输的二进制码。

该过程实验代码如下：

代码 3-1 信源信息获取

```
1 %% 信源信息
2 img=imread('logo.jpg');
3 img=rgb2gray(img);
4 level=graythresh(img); %使用graythresh计算灰度阈值
5 bimg=imbinarize(img,level);
6 soure=reshape(bimg,1,[]);
```

由于 MATLAB 中数据均为离散数据，所以要设定每个码元的采样数，以及码元的持续时间，其基本参数设置如下：

代码 3-2 基本参数

```
1 %% 基本参数
2 M=length(wave); % 产生码元数
3 L=100; % 每码元复制L次,每个码元采样次数
4 Ts=0.001; % 每个码元的宽度,即码元的持续时间
5 Rb=1/Ts; % 码元速率1K
6 dt=Ts/L; % 采样间隔
7 TotalT=M*Ts; % 总时间
8 t=0:dt:TotalT-dt; % 时间
9 TotalT2=(M/2)*Ts; % 总时间2
10 t2=0:dt:TotalT2-dt; % 时间2
11 Fs=1/dt; % 采样间隔的倒数即采样频率
12 wave=soure;
```

之后，我们将位于奇数位置的码作为 I 路输入，位于偶数位置的码作为 Q 路输入。本实验中，我们采用 $N = 1$ 的编码映射，即 I 路码和 Q 路码输入 0，对应映射输出为 $1/\sqrt{2}$ ，输入 1，对应映射输出为 $-1/\sqrt{2}$ ；其中 L 个码元重复代表该码元的持续时间，代码实现如下：

代码 3-3 I、Q 路码元

```
1 %% I、Q路码元
2 wave=1-2*wave; % 单极性变双极性
3 fz=ones(1,L); % 定义复制的次数L,L为每码元的采样点数
4 % I路码元是基带码元的奇数位置码元，Q路码元是基带码元的偶数位置码元
5 I=zeros(1,M/2);Q=zeros(1,M/2);
6 for i=1:2:M
7     I((i+1)/2)=wave(i);
```

```

8 end
9 for i=2:2:M
10     Q(i/2)=wave(i);
11 end
12 x2 = I(fz,:); % 将原来I的第一行复制L次, 称为L*(M
    /2)的矩阵
13 I_lu = reshape(x2,1,L*(M/2)); % 将刚得到的L*(M/2)矩阵, 按列重新排列
    形成1*(L*(M/2))的矩阵
14 x3 = Q(fz,:); % 将原来Q的第一行复制L次, 称为L*(M
    /2)的矩阵
15 Q_lu = reshape(x3,1,L*(M/2)); % 将刚得到的L*(M/2)矩阵, 按列重新排列
    形成1*(L*(M/2))的矩阵

```

接下来进行 QPSK 调制, 设置载波频率为 2kHz, 则只需要将得到的 I 路信号和 Q 路信号与载波进行点乘即可。需要注意的是 MATLAB 中数据都是离散的, 所以运算需要用点乘。具体代码如下:

代码 3-4 QPSK 调制

```

1 %% QPSK调制
2 fc=2000; % 载波频率2kHz
3 zb1=cos(2*pi*fc*t2); % 载波1
4 psk1=I_lu.*zb1; % PSK1的调制
5 zb2=sin(2*pi*fc*t2); % 载波2
6 psk2=Q_lu.*zb2; % PSK2的调制
7 qpsk=psk1+psk2; % QPSK的实现

```

信号通过一个高斯白噪声信道, 再 MATLAB 中利用 awgn 函数实现即可, 代码如下:

代码 3-5 信号经过高斯白噪声信道

```

1 %% 信号经过高斯白噪声信道
2 tz=awgn(qpsk,0); % 信号qpsk中加入白噪声, 信噪比为SNR
    =20dB

```

信息传输完毕, 接下来进行信号的解调。首先是相干解调, 先乘以相干载波:

代码 3-6 解调部分

```

1 %% 解调部分
2 tz1=tz.*zb1; % 相干解调, 乘以相干载波
3 tz2=tz.*zb2; % 相干解调, 乘以相干载波

```

在经过低通滤波器, 在 MATLAB 中信号经过滤波器可以利用函数 ftfilt 实现, 滤波器的设计采用 FIR 滤波器, 生成一个 31 抽头系数的矩形窗滤波器, 实现代码如下:

代码 3-7 加噪信号经过滤波器

```

1 %% 加噪信号经过滤波器
2 % 低通滤波器设计
3 fp=2*Rb; % 低通滤波器截止频率, 乘以2是因为下面
    要将模拟频率转换成数字频率wp=Rb/(Fs/2)

```

```

4 b=fir1(30, fp/Fs, boxcar(31));% 生成fir滤波器系统函数中分子多项式
   的系数
5 % fir1函数三个参数分别是阶数, 数字截止频率, 滤波器类型
6 % 这里是生成了30阶(31个抽头系数)的矩形窗滤波器
7 [h,w]=freqz(b, 1,512);      % 生成fir滤波器的频率响应
8 % freqz函数的三个参数分别是滤波器系统函数的分子多项式的系数, 分母多项式
   的系数(fir滤波器分母系数为1)和采样点数(默认)512
9 lvbo1=fftfilt(b,tz1);      % 对信号进行滤波, tz1是等待滤波的信
   号, b是fir滤波器的系统函数的分子多项式系数
10 lvbo2=fftfilt(b,tz2);      % 对信号进行滤波, tz2是等待滤波的信
   号, b是fir滤波器的系统函数的分子多项式系数

```

接下来分别对 I 路和 Q 路信号进行判决, 两个过程分别等价于两个独立的 BPSK 判决过程, 但为了与我们映射相对应, 将滤波后的向量的每个元素和 0 进行比较, 大于 0 则为 0, 小于 0 则为 1, 这是因为我们最开始对信息码元做了变换映射, 所以得到的大于 0 的对应 0, 小于 0 的对应 1, 具体代码如下:

代码 3-8 抽样判决

```

1 %% 抽样判决
2 k=0;                      % 设置抽样限值
3 pdst1=1*(lvbo1<0);      % 滤波后的向量的每个元素和0进行比较, 大
   于0为1, 否则为0
4 pdst2=1*(lvbo2<0);      % 滤波后的向量的每个元素和0进行比较, 大
   于0为1, 否则为0

```

最后进行一个将 I 路和 Q 路码进行合并, 在将多余的抽样码舍弃, 即可得到我们的传输的信源码:

代码 3-9 I、Q 路合并

```

1 %% I、Q路合并
2 I_zong = [];
3 Q_zong = [];
4 % 取码元的中间位置上的值进行判决
5 for j=L/2:L:(L*M/2)
6     if pdst1(j)>0
7         I_zong=[I_zong,1];
8     else
9         I_zong=[I_zong,0];
10    end
11 end
12 % 取码元的中间位置上的值进行判决
13 for k=L/2:L:(L*M/2)
14     if pdst2(k)>0
15         Q_zong=[Q_zong,1];
16     else
17         Q_zong=[Q_zong,0];
18     end
19 end
20 code = [];

```



```
21 % 将I路码元为最终输出的奇数位置码元，将Q路码元为最终输出的偶数位置码元
22 for n=1:M
23     if mod(n, 2)~=0
24         code = [code, I_zong((n+1)/2)];
25     else
26         code = [code, Q_zong(n/2)];
27     end
28 end
```

第 4 章 仿真结果

4.1 10dB 高斯信道传输结果

绘制信源和 QPSK 调制后的频谱以及 I 路和 Q 路乘以相干载波后的频谱结果如下：

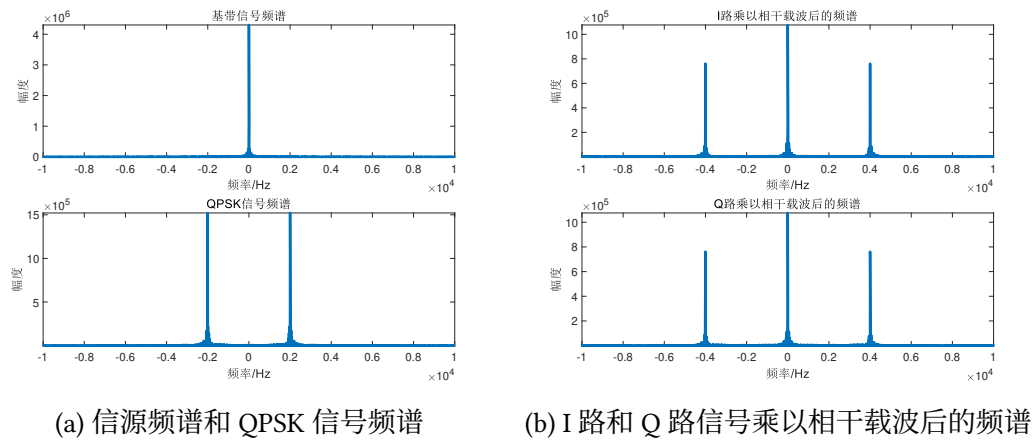


图 4-1 频谱图

所设计的 FIR 滤波器频谱和经过滤波器后得到 IQ 两路信号频谱如图 4-2：

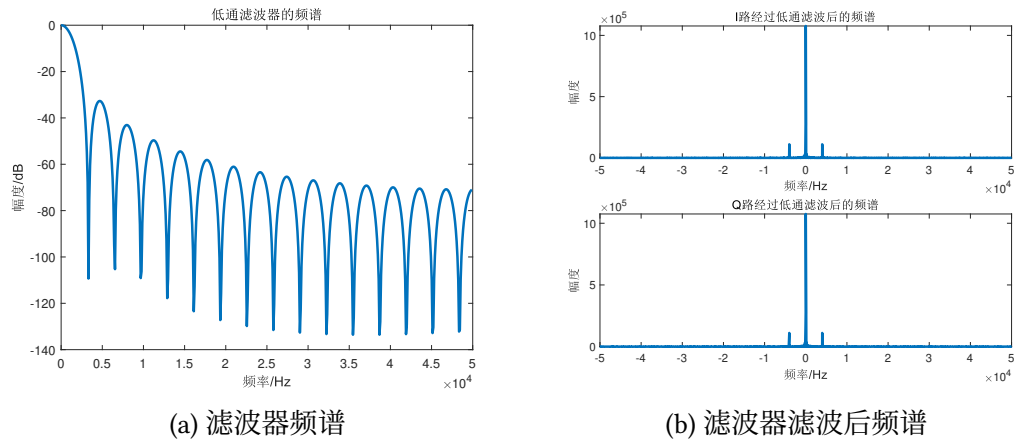


图 4-2 滤波器与滤波结果

最后得到通过高斯信道前的星座图和通过高斯信道后的星座图如图 4-3 所示：

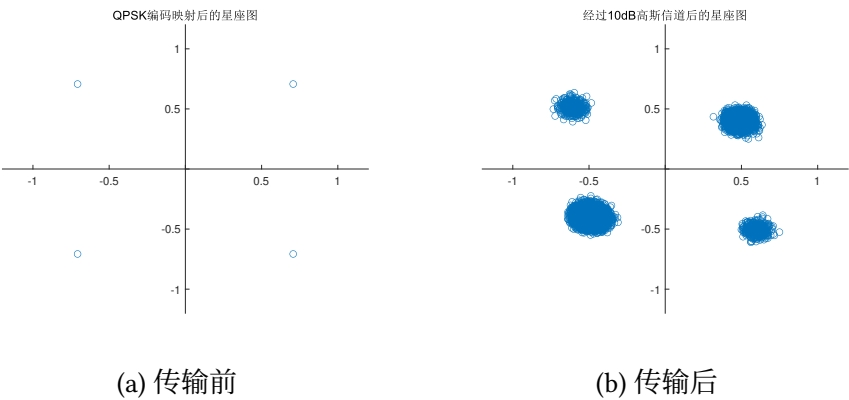


图 4-3 信号星座图

最终得到传输前后的图像效果如图 4-6 所示，可以发现两者没有区别，计算可得传输误比特率为 0。

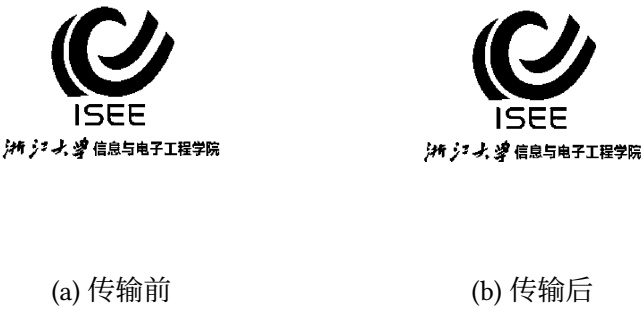


图 4-4 传输图片

4.2 误码率统计结果

取 SNR 范围为-5dB 至 15dB，得到 SNR 与误比特率 BER 的曲线如图 4-7 所示，其中 0dB，5dB，10dB，15dB 对应误比特率如表 4-1 所示，可以发现仿真的误比特率显著小于理论值。其原因是传输的数据 0 和 1 并不是等概率的发送，并且发送的数据存在一定的规律性，如长时间传 0 或长时间传 1，而理论计算是基于 0 和 1 等概率发送的情况下得到的结果，所以仿真值会小于理论值。

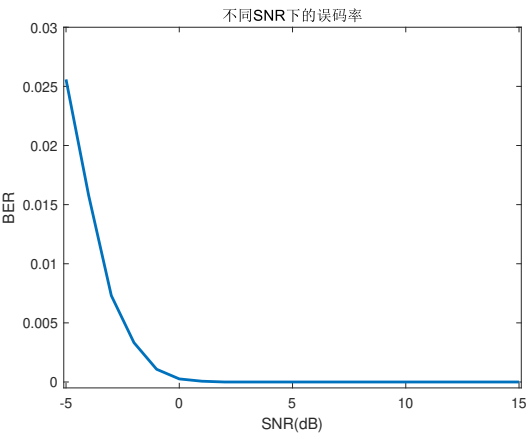


图 4-5 不同信噪比下误码率

表 4-1 不同信噪比下误码率

SNR	SNR=0dB	SNR=5dB	SNR=10dB	SNR=15dB
BER(仿真)	$2.609 \cdot 10^{-4}$	0	0	0
BER(理论)	0.1587	$8.16 \cdot 10^{-4}$	0	0

4.3 不同信噪比接收图像

接收信噪比为 0dB, 5dB, 10dB, 15dB 的接收图像如图 4-8 所示，可以发现除了 0dB 的情况，其他信噪比下图像没有失真，0dB 则出现了一定的噪点，但图像没有严重的畸变或失真。



图 4-6 不同信噪比接收图像