

实验二报告 连续波 MATLAB 实现

夏厚 PB18051031

2021 年 5 月 10 日

1 实验目的

- 掌握线性模拟调制信号的波形及产生方法；
- 掌握线性模拟调制信号的频谱特点；
- 掌握线性模拟调制信号的解调方法；
- 掌握线性模拟调制系统的 MATLAB 仿真实现。

2 实验原理

2.1 模拟调制

模拟调制包括幅度调制 (DSB,SSB,AM) 和相角调制 (频率和相位调制), 其中幅度调制 (线性调制) 是正弦载波的幅度随着调制信号而改变的调制方案。若调制信号 $m(t)$, 频谱为 $M(f)$, 带宽为 $B = f_m$, SSB 调制的带宽为 B , DSB 和 AM 调制的带宽为 $2B$ 。VSB-AM 的带宽在 $B-2B$ 区间内。

AM 调制与 DSB 调制在许多方面十分相似, 唯一的区别在于 AM 调制用 $[A + m(t)]$ 代替 DSB 的 $m(t)$ 。AM 调制信号可以表示为:

$$s_{AM}(t) = [A + m(t)]\cos(2\pi f_c t)$$

该调制信号的频谱可以表示为:

$$S_{AM}(f) = \frac{A_c}{2} [\delta(f + f_c) + \delta(f - f_c) + aM(f + f_c) + aM(f - f_c)]$$

解调器的输入信噪比定义为：

$$r_i = \frac{S_i}{N_i} = \frac{S_i}{n_0 B_{BPF}}$$

其中， B_{BPF} 为理想带通滤波器带宽，在理想信道中，当带通滤波器的幅频特性为常数 1 时， $\frac{S_i}{N_i} = \frac{S_i}{n_0 B_{BPF}}$ ， $n_0 = \frac{P_s}{t_i B_{BPF}}$ ，即可由解调器输入信噪比计算出信道噪声的单边带功率谱密度，当系统抽样速率为 f_s 时，产生的高斯白噪声带宽为 $f_s/2$ ，由此可以计算出信道中高斯白噪声的平均功率，即方差 $\sigma_n^2 = n_0 f_s/2$ ，可以产生信道中叠加的高斯白噪声。

2.2 信道加性高斯白噪声

信道中加性高斯白噪声功率由于其均值为 0，故其方差 $\sigma^n = n_0 B$ 即为其平均功率，其中 n_0 为高斯白噪声单边带功率谱密度；B 为信道带宽。

在理想通信系统中，利用已给解调器输入信噪比及已调信号功率和带宽，可以计算出 n_0 ，从而算出信道加性高斯白噪声的方差，由于其均值为 0，故该方差为其平均功率，利用它可以生成信道加性高斯白噪声。转换关系为：

$$r = \frac{S_i}{N_i} = \frac{S_i}{n_0 B_{BPF}}, n_0 = \frac{S_i}{r B_{BPF}}, \sigma^2 = \frac{n_0 f_s}{2} = \frac{S_i f_s}{2r B_{BPF}}$$

$$\sigma \sqrt{\frac{S_i f_s}{2r B_{BPF}}} \text{ 或 } \sigma^2 = \frac{E f_s}{4r}, \text{ E 是接收信号平均能量。}$$

3 实验内容

3.1 产生调幅 (AM) 信号，并观察对其时域频域特征

编写函数 `mod_AM` 用于产生 AM 调制信号，其中输入参数为被调制信号 x ，调制系数 k ，载波 f_c ，输出参数为调制信号 y 。

$$y(t) = A_c[1 + kx(t)]\cos(2\pi f_c t)$$

完成该实验内容分为如下步骤：

- 在主程序中产生一个将被调制的信号，观察其时频特性
- 编写 AM 调制函数 `mod_AM(x, k, f_c)`，其中载波信号在函数中给出

- 在主函数中调用函数 $mod_AM(x, k, f_c)$, 对信号进行调制, 并输出 AM 调制信号的时频特性

实验中用到的调制信号为,

$$m(t) = \begin{cases} 1, & 0 \leq t \leq t_0/3 \\ -2, & t_0/3 < t \leq 2t_0/3 \\ 0, & else \end{cases}$$

图形化输出结果见图 1, 代码如下:

```

1      clear all; close all; echo on
2      %参数设定
3      A=3;
4      fc=250;
5      t0=0.15;
6      snr=25;
7      dt=0.001;
8      fs=1/dt;
9      df = 0.2;
10     t=0:dt:t0;
11     Lt=length(t);
12     snr_lin = 10^(snr/10);
13     %产生被调制信号
14     m=[ones(1,50), -2*ones(1,50), zeros(1,50+1)];
15     L=2*min(m);
16     R=2*max(abs(m))+A;
17     %输出被调制信号时频特性
18     figure(1); subplot(421);
19     plot(t, m(1:length(t)));
20     axis([0 t0 -R/2 R/2]);
21     xlabel('t');
22     ylabel('调制信号');
23     subplot(422);
24     [M, m, df1, f]=T2F(m, dt, df, fs);
25     [Bw_eq]=signalband(M, df, t0);

```

```

26         f_start=fc-Bw_eq;
27         f_cutoff=fc+Bw_eq;
28         plot(f,fftshift(abs(M)));
29         xlabel('f');ylabel('调制信号频谱');
30         %对信号进行调制并输出时频特性
31         subplot(423);
32         u=mod_AM(m,1,fc);%AM调制
33         plot(t,u);
34         axis([0 t0 -R R]);
35         xlabel('t');ylabel('AM已调信号');
36         subplot(424);
37         [U,u,df1,f]=T2F(u,dt,df,fs);
38         plot(f,fftshift(abs(U)));
39         xlabel('f');ylabel('AM已调信号频谱');

```

AM 调制函数 $mod_AM(x, k, f_c)$:

```

1         function y=mod_AM(x,k,fc)
2         A=3;
3         t0=0.15;
4         dt=0.001;
5         t=0:dt:t0;
6         Lt=length(t);
7         c=cos(2*pi*fc*t);%载波信号
8         y=(A+k*x(1:Lt)).*c(1:Lt);%对信号进行AM调制

```

3.2 产生调频 (FM) 信号，并观察对其时频域特征

编写函数 mod_FM 用于产生 FM 调制信号，其中输入参数为被调制信号 x ，调制系数 b ，载波频率 f_c ，输出参数为调制信号 y 。

$$y(t) = A_c \cos[2\pi f_c t + b \int_0^t x(\tau) d\tau]$$

完成该实验内容分为如下步骤：

- 编写 FM 调制函数 $mod_FM(x, b, f_c)$ ，其中载波信号在函数中给出

- 利用实验内容 1 中产生的被调制信号，调用 $mod_FM(x, b, f_c)$ 对其进行 FM 调制
- 输出调制后信号的时频特性

图形化输出结果见图 1，代码如下：

```

1      %对信号进行FM调制并输出调制后信号时频特性
2      subplot(425);
3      v=mod_FM(m,1 , fc );%FM
4      plot(t , v);
5      axis([0 t0 -R R]);
6      xlabel('t');ylabel('FM已调信号');
7      subplot(426);
8      [U,u, df1 , f]=T2F(v, dt , df , fs );
9      plot(f , fftshift(abs(U)));
10     xlabel('f');ylabel('FM已调信号频谱');

```

FM 调制函数 $mod_FM(x, b, f_c)$:

```

1      function y=mod_FM(x , b , fc )
2      A=3;
3      t0=0.15;
4      dt=0.001;
5      t=0:dt:t0;
6      Lt=length(t);
7      fm=zeros(1 , Lt);
8      %积分离散为累加
9      for p=1:Lt
10         for q=1:p
11             fm(p)=x(q)+fm(p);
12         end
13     end
14     y=A.*cos(2*pi*fc*t+b*fm);%对信号进行FM调制

```

3.3 产生调相 (PM) 信号，并观察对其时频域特征

编写函数 `mod_PM` 用于产生 PM 调制信号，其中输入参数为被调制信号 x ，调制系数 k ，载波频率 f_c ，输出参数为调制信号 y 。

$$y(t) = A_c \cos[2\pi f_c t + kx(t)]$$

该实验步骤与实验内容 2 步骤同理

图形化输出结果见图 1，代码如下：

```
1      %对信号进行PM调制并输出调制后信号时频特性
2      subplot(427);
3      m=[ones(1,50),-2*ones(1,50),zeros(1,50+1)];
4      g=mod_PM(m,1,fc);%PM
5      plot(t,g);
6      axis([0 t0 -R R]);
7      xlabel('t');ylabel('PM已调信号');
8      subplot(428);
9      [U,u,df1,f]=T2F(g,dt,df,fs);
10     plot(f,fftshift(abs(U)));
11     xlabel('f');ylabel('PM已调信号频谱');
```

PM 调制函数 `mod_PM(x,k,fc)`:

```
1      function y=mod_PM(x,k,fc)
2      A=3;
3      t0=0.15;
4      dt=0.001;
5      t=0:dt:t0;
6      Lt=length(t);
7      y=A.*cos(2*pi*fc*t+k.*x);%对信号进行PM调制
```

分析图 1 实验结果：

观察调制信号的频域特性，由于该调制信号时域具有阶跃性变化，所以可以看到，该信号频域无限展宽。在 MATLAB 中对模拟信号进行处理时，我们会对其进行采样离散化，这也是为什么调制信号时域中的“跳变”成为斜线而不是竖直线。

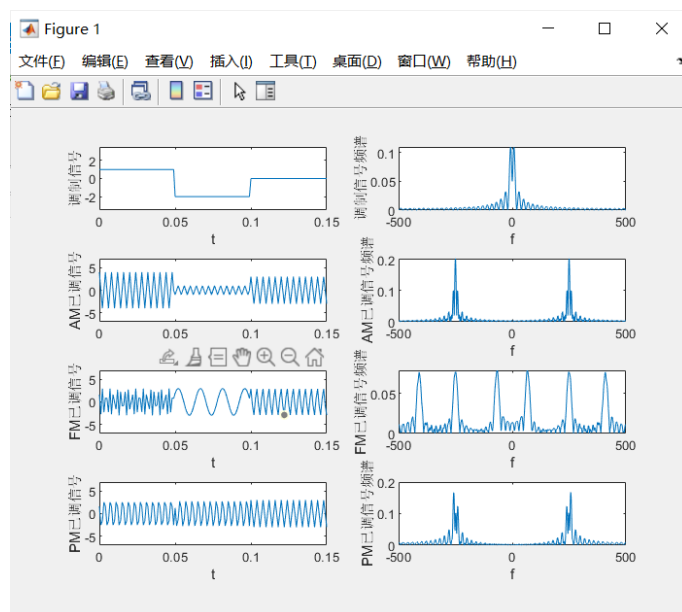


图 1: 被调制信号, AM、FM、PM 调制后信号的时频特性

实验中使用的 AM 调制格式为 $y(t) = [A + kx(t)]\cos(2\pi f_c t)$, 其中 $A=3$, $k=1$, 调制信号第一阶段值为 1, 第二阶段值为-2, 第三阶段值为 0. 观察 AM 调制后信号时域特性, 第一阶段 $A+1=4$, 所以幅度调制到最大值为 4, 第二阶段 $A-2=1$, 所以幅度调制到最大值为 1, 第三阶段 $A+0=3$, 所以幅度调制到最大值为 3. 观察其频域特性, 调制后信号频谱是原信号频谱的平移并叠加两个冲击频谱, 这是因为调制信号 $\cos(2\pi f_c t)$ 带来原信号频谱的搬移, 直流分量 A 带来两个冲击频谱. 综上所述, AM 调制实验结果与理论分析一致。

观察 FM 调制结果的时域特性, 因为调制的是频率, 所以在图像中可以清晰看到调制后信号第一阶段频率明显高于后两个阶段, 第二阶段频率最小, 这是由于调制信号在第二阶段函数值为-2. 图像中调制后信号第一阶段的幅度有起伏, 按照 FM 调制原理, 幅度本应没有变化, 仔细查看代码后, 发现原因是在信号采样离散时采样频率较小, 造成此处的信号失真. 观察频域特性, 存在三个主频率, 分别由三个阶段的频率调制带来. 综上所述, FM 调制实验结果与理论分析一致。

观察 PM 调制结果时域特性, 在三个阶段的交界处, 信号的相位发生跳变. 因为是相位调制, 所以而频率相同. 从频域特性中也能看出, AM 调

制与 PM 调制的频域特性具有相同的中心频率。综上, PM 调制实验结果与理论分析一致。

3.4 解调 AM 调制信号

编写函数 *dem_AM* 用于解调 AM 调制信号, 其中输入参数为调制信号 *y*, 调制系数 *k*, 载波 f_c , 输出参数为调制信号 *x*。完成该实验内容分为如下步骤:

- 同实验内容 1 的调制信号, 先进行 AM 调制
- 将调制信号叠加高斯噪声, 其中高斯噪声由实验原理中计算方法得到
- 对叠加高斯噪声后的信号, 输入解调函数中进行解调

实验结果图像化输出见图 2, 代码如下:

```
1      %产生噪声并输出噪声时频特性
2      signal_power = power_x(u(1:Lt));
3      noise_power=(signal_power*fs)/(snr_lin*4* Bw_eq);
4      noise_std = sqrt(noise_power);
5      noise = noise_std*randn(1,Lt);
6      figure(2); subplot(321);
7      plot(t,noise);
8      axis([0 t0 -R R]);
9      xlabel('t');ylabel('噪声信号');
10     subplot(322);
11     [noisef,noise,df1,f]=T2F(noise,dt,df,fs);
12     plot(f,fftshift(abs(noisef)));
13     xlabel('f');ylabel('噪声频谱');
14     sam=u(1:Lt)+noise(1:Lt);
15     subplot(323);
16     plot(t,sam);
17     axis([0 t0 -R R]);
18     xlabel('t');
19     ylabel('加噪的信号');
20     subplot(324);
```



```

21     [samf,sam,df1,f]=T2F(sam,dt,df,fs);
22     plot(f,fftshift(abs(samf)));
23     xlabel('f');
24     ylabel('加噪信号频谱');
25     [H,f]=bp_f(length(sam),f_start,f_cutoff,df1,fs,1);
26     sam=u(1:Lt)+noise(1:Lt);
27
28     x=dem_AM(sam,1,fc);%解调AM
29     subplot(325);
30     plot(t,x);
31     axis([0 t0 -2 2]);
32     xlabel('t');ylabel('AM解调信号');
33     subplot(326);
34     [U,u,df1,f]=T2F(x,dt,df,fs);
35     plot(f,fftshift(abs(U)));
36     xlabel('f');ylabel('AM解调信号频谱');

```

AM 解调函数 $dem_AM(y,k,f_c)$:

```

1     function x=dem_AM(y,k,fc)
2     A=3;
3     t0=0.15;
4     dt=0.001;
5     t=0:dt:t0;
6     Lt=length(t);
7     x1=y.*cos(2*pi*fc*t);%相干
8     f_cutoff=250;
9     x2=lowpass(x1,f_cutoff,1000);%通过低通滤波器
10    x=abs(x2)-mean(abs(x2));%去除直流分量

```

分析图 2 实验结果

产生的随机噪声，在时域和频域都具有混乱性。叠加噪声的 AM 调制后的信号，在时域和频域与原 AM 调制信号的时频特性相似，这是由于噪声的大小相对很小。观察解调之后的信号，在时域中仍然有小量误差波动，这是小量随机噪声相干并解调带来的。仔细观察时域特性的幅度，发现第一阶段与第二阶段大小不是 1 和-2，这是由于在解调函数 $dem_AM(y,k,f_c)$

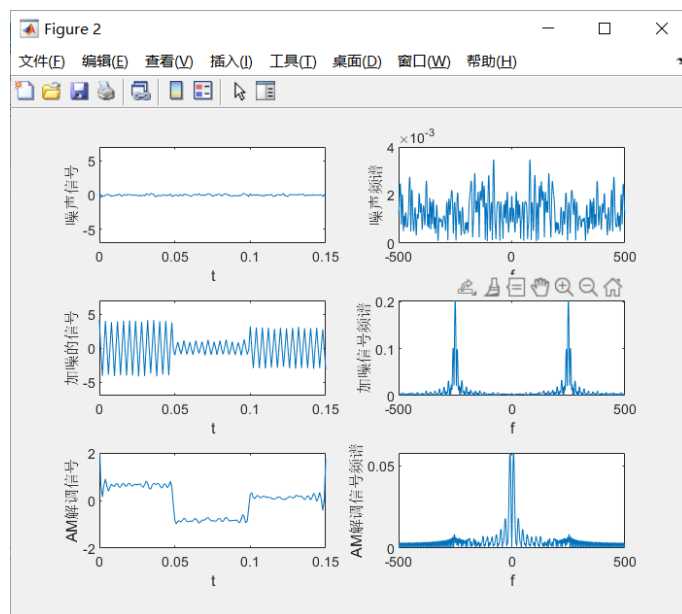


图 2: 噪声信号、叠加噪声的 AM 调制信号、AM 解调的时频特性

中对解调系数 k 的使用的原因，为方便我直接使用了 MATLAB 内置低通函数 `lowpass`，信号各阶段的相对关系依然成立，并不会影响对调制信号的判决。解调后信号因为通过了一个低通滤波器，所以高频成分较小，在其频域特性中也能看到，而整体结果和原调制信号近似。

4 总结

- 计算机对模拟信号进行分析时，需要对信号进行采样，然后在离散域上进行处理。采样过程需要注意采样率不能过低。对应到 MATLAB 中，就是一个长度的信号，用多少个点（步长）来表示。
- 因为本实验要求分别编写 AM、FM、PM 调制函数，通过本实验进一步熟悉了 MATLAB 编程函数的格式、路径和调用。