CPM信号的非相干序列检测

一、背景知识

1.1相干检测和非相干检测

接收信号时，有相干检测和非相干检测两种方法。对于接收到的信号$S(t) = Acos(\omega_c t + \phi) + N(t)$和载波$cos(\omega_c t + \phi')$，相干检测可以通过锁相环(phase lock loop)等方式使得$\Delta\phi = \phi - \phi' \to 0$

这样恢复的信号就是$Y_k = X_k + N_k$

相干检测的解调效果较好，但是会有成本高等问题。而对于非相干检测，没有相位恢复这个过程，即$\Delta\phi = \phi - \phi' \neq 0$

此时就不能单纯地用相干检测的方法恢复信号，因为此时得到的是$Y_k = X_k e^{j\theta} + N_k$，而我们并不知道相位信息。如何将非相干检测的质量逼近相干检测，是一个很重要的问题[1]。

1.2CPM信号

连续相位调制(continuous phase modulation)，即CPM，是一种重要的数学调制方式。不同于在每个符号发射时载波相位发生突变的调制方式，CPM载波的相位一直保持连续的状态。正因如此其可产生高频谱效率，它的主要缺点是最优接收器需要很高的实现复杂性[2]。

在论文《Noncoherent Sequence Detection of Continuous Phase Modulations》中，作者基于洛朗分解，将非相干检测应用在CPM调制信号中；并通过白化匹配滤波等方式，使非相干检测的结果非常逼近相干检测，且能够控制住复杂度。论文的整体框架如下：

以下将从信号发射、传输、接收三大方面介绍工作内容和思路。

二、信号发射

在发射端，需要进行CPM信号的生成与洛朗分解。

2.1信号生成

CPM信号的复包络有如下形式：

$$s(t, \overrightarrow{a}) = \sqrt{\frac{2E_s}{T}} exp\{j2\pi h \sum_n a_n q(t - nT)\}$$

其中E_s是每个符号的能量；T是符号周期；h是调制系数(modulation index)，$h = k/p$，k、p互质；$\{a_n\}$是需要传输的符号，取值自M-ary表$\{\pm1, \pm3, ..., \pm(M-1)\}$，$M$取偶数；$q(t)$是相位平滑相应函数(phase-smoothing response)，其导数$g(t)$是频率脉冲函数(frequency pulse)。$q(t)$有如下性质：

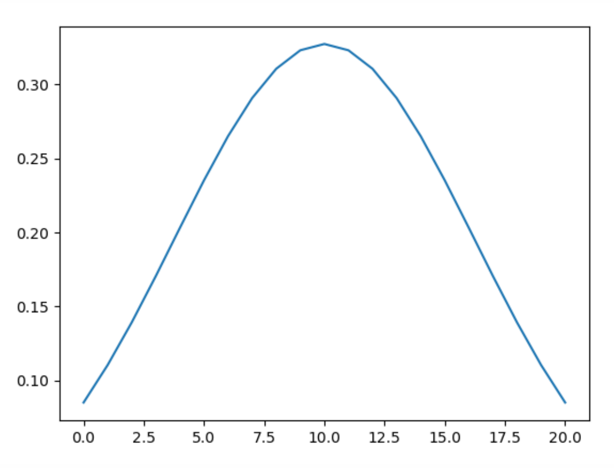
$$q(t) = \begin{cases} 0, & t < 0 \\ \int_0^\tau g(\tau)d\tau, & 0 \le t \le LgT \\ \frac{1}{2}, & t > LgT \end{cases}$$

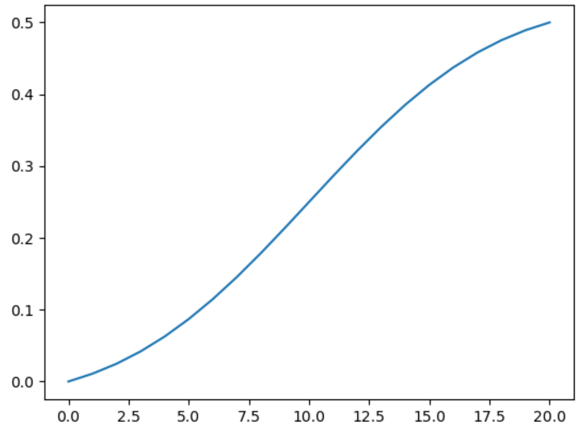
对于GMSK信号，有

$$g(t) = \begin{cases} \frac{1}{2T}[Q(2\pi B\frac{t-\frac{T}{2}}{\sqrt{ln2}}) - Q(2\pi B\frac{t+\frac{T}{2}}{\sqrt{ln2}})], & -\frac{LgT}{2} \le t \le \frac{LgT}{2} \\ 0, & otherwisw \end{cases}$$

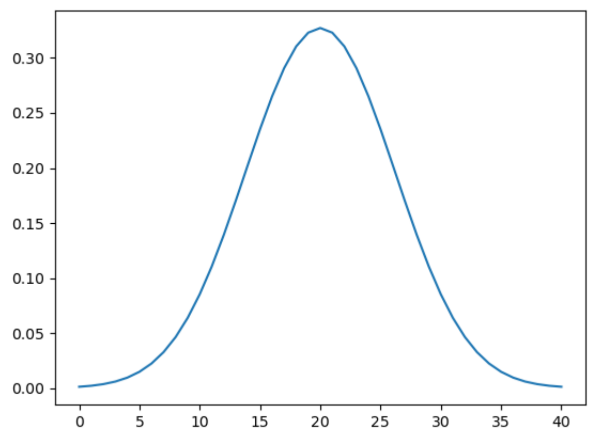
注意到，$q(t)$只有在$[0, LgT]$时是$g(t)$的积分，所以在实际操作时，需要将$g(t)$向右平移$\frac{LgT}{2}$；此外，我们可以观察到$g(t)$在全空间积分才是$\frac{1}{2}$，即$q(LgT)$并不是严格的$\frac{1}{2}$，所以在实际操作中，我将$q(t)$进行了归一化处理，使其满足$q(LgT) = \frac{1}{2}$。

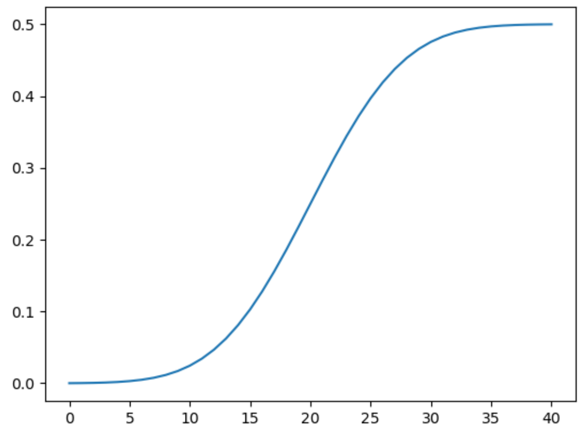
当$Lg = 2$时，频率脉冲和相位相应函数如下：





当$Lg=4$时，频率脉冲和相位相应函数如下：



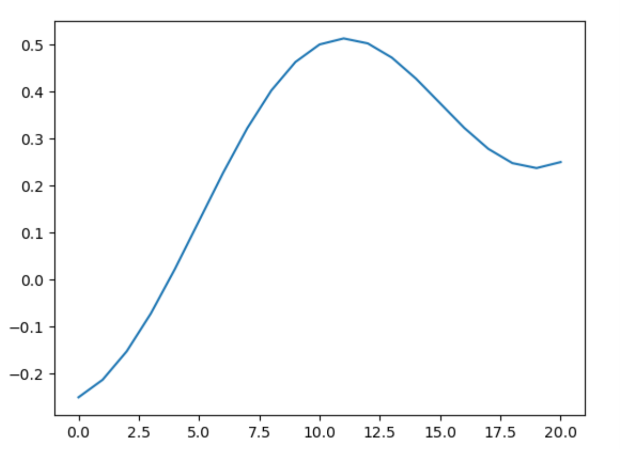


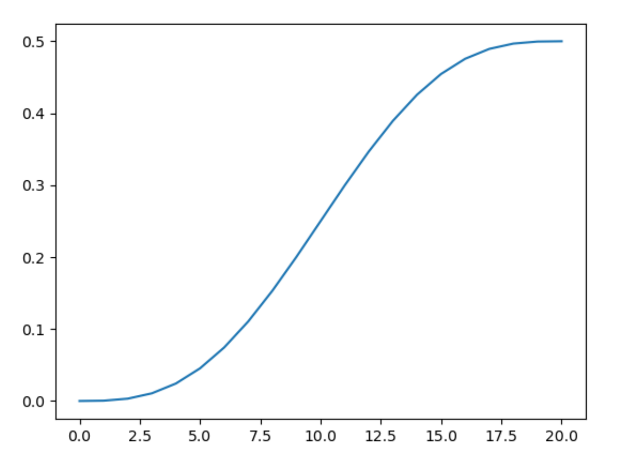
观察到随着Lg的增大，频率脉冲的能量越来越向中间“集中”。

对于2RC信号，有：

$$g(t)=\begin{cases}\frac{1}{2LgT}[t-cos(\frac{2\pi t}{LgT})], & 0\leq t\leq LgT\\ 0, & otherwisw\end{cases}$$

当$Lg=2$时，其频率脉冲和相位相应函数如下：





知道$q(t)$，我们就可以对序列$\{a_n\}$进行连续相位调制，将得到的$s(t, \overrightarrow{a})$进行发送。

2.2洛朗分解

数学上可以严谨证明，CPM信号可以分解成一系列PAM波的叠加：

$$s(t, \overrightarrow{a}) = \sum_{k=0}^{Q^{log_2 M}(M-1)-1} \sum_{n} \alpha_{n,k} h_k(t - nT)$$

其中$Q = 2^{Lg-1}$，以二元信号(M=2)为例，上式化简为：

$$s(t, \overrightarrow{a}) = \sum_{k=0}^{Q-1} \sum_{n} \alpha_{n,k} h_k(t - nT), \quad 0 \leq k \leq Q - 1$$

脉冲$h_k(t)$用如下方法得到：

$$h_k(t) = \prod_{i=0}^{Lg-1} u(t + iT + \beta_{k,i} LT) \quad 0 \leq k \leq Q - 1$$

$$u(t) = \begin{cases} sin[2\pi hq(t)]/sin(h\pi), & 0 \leq t \leq LgT \\ u(2LgT - t), & LgT < t \leq 2LgT \\ 0, & otherwisw \end{cases}$$

$\beta_{k,i}$由k的二进制分解得到：

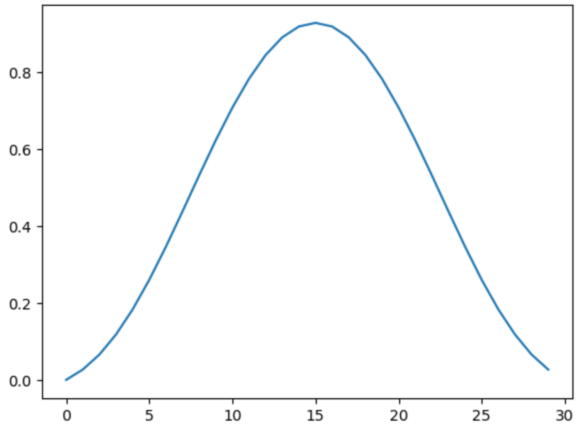
$$k = \sum_{i=1}^{Lg-1} 2^{i-1} \beta_{k,i}$$

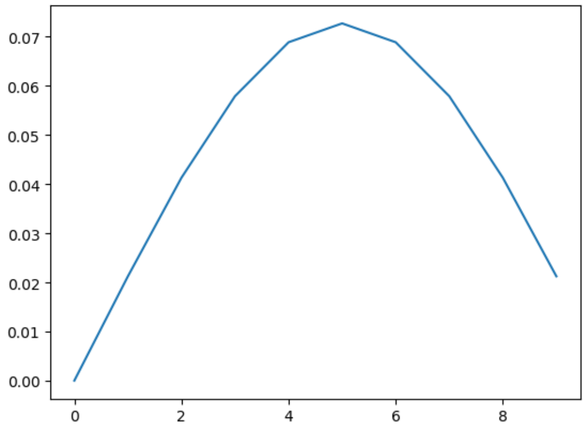
$h_k(t)$的持续时间由D_k决定：

$$D_k = min\{Lg(2-\beta_{k,i})-i\}, \quad 0 \le i \le Lg-1$$

实际场景中可进行优化降低复杂度，详见第六部分。

GMSK信号的$h_0(t)$和$h_1(t)$如下图所示：





伪符号序列$\alpha_{n,k}$由以下公式得到：

$$\alpha_{n,k} = exp\{jh\pi[\sum_{m=-\infty}^{n} a_m - \sum_{i=0}^{Lg-1} a_{n-i}\beta_{k,i}]\}$$

实际场景中，我们需要推导出$\alpha_{k,n}$的迭代公式，如对于GMSK信号K=1的情况下有：

$$\alpha_n = ja_n\alpha_{n-1}$$

K=2时有：

$$\alpha_{0,n} = ja_n\alpha_{0,n-1}$$
$$\alpha_{1,n} = ja_n\alpha_{1,n-1}$$

当M>2的多元符号时，推导比较复杂，详见附录一。

至此我们得到了CPM调制信号$s(t,\overrightarrow{a})$、脉冲$h_k(t)$、序列$\alpha_{n,k}$，注意我们唯一需要发送的只有CPM信号，$h_k(t)$和$\alpha_{n,k}$在解调时才会用到。

三、信号传输

通过信道传输后，接收到的信号形式为：

$$r(t)=s(t,\overrightarrow{a})e^{j\theta}+w(t)$$

θ服从于$[0,2\pi]$的均匀分布，$w(t)$是功率谱密度为n_0的高斯白噪声。我们需要在θ未知的情况下，成功解调出序列$\{a\}$，并使信噪比尽可能逼近相干序列检测。

四、信号接收

在接收端，接收信号要依次经过匹配滤波、白化滤波和维特比解码。该部分的整体框图如下：

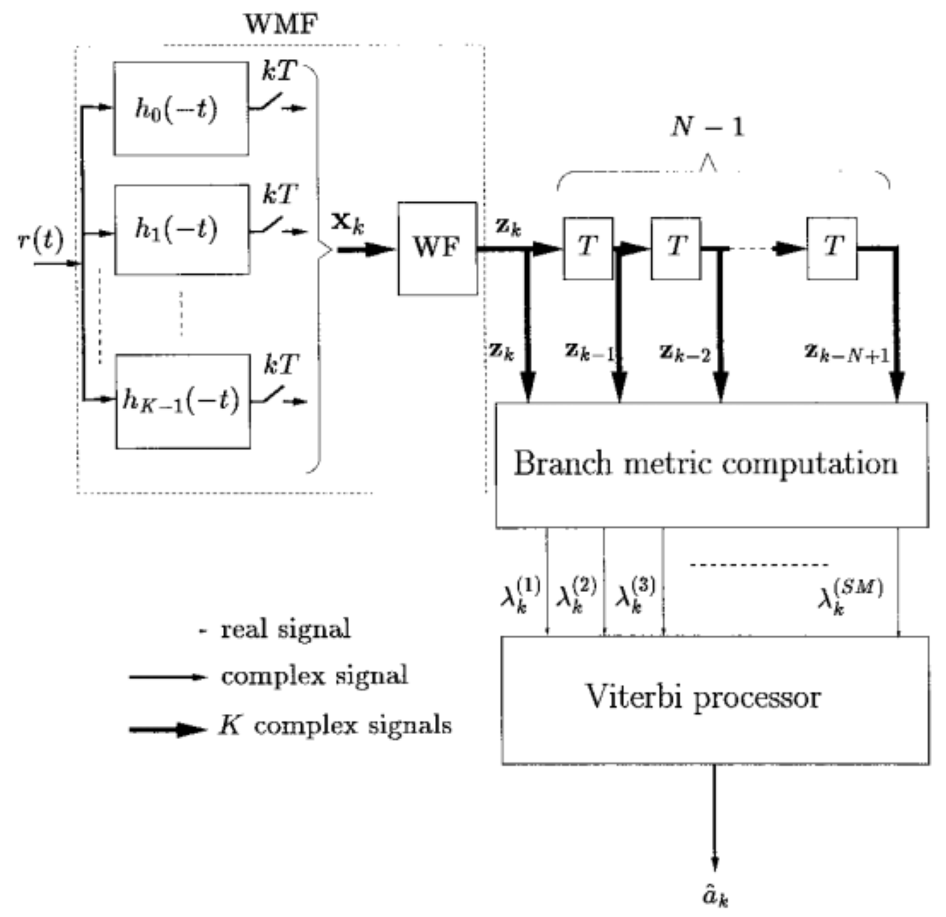


Fig. 1. Noncoherent sequence detection receiver for CPM.

4.1匹配滤波

即将$r(t)$通过我们课上所学的匹配滤波器得到$x_{k,n}$：

$$x_{k,n}=r(t)\otimes h_k(-t)|_{t=nT}$$

4.2白化滤波

从先前的工作来看，直接将得到的$x_{k,n}$解调得到的效果很差，其原因在于此时的噪声项并不是规则的高斯白噪声，证明如下：

$$x_{k,n} = r(t) \otimes h_k(-t)|_{t=nT} = s_{k,n}e^{j\theta} + n_{k,n}$$

$$n_{k,n} = w(t) \otimes h_k(-t)|_{t=nT}$$

$$g_{m,k}(t) = h_m(t) \otimes h_k(t)$$

则噪声项的自相关函数为$\overrightarrow{R}_n(m) = 2N_0 \overrightarrow{G}_m^T$，谱密度矩阵为$\Phi_n(z) = 2N_0 G^T(z)$，显然此时的噪声并不是白噪声，影响了解调，于是该论文提出了白化滤波，将噪声白化后大大提升了解调质量。

在CPM信号的调制解调中，矩阵$\Phi_n(z)$沿着单位圆在大多数情况下都是正定的，即使在一些病态的情况下，比如2RC调制中的$h_1(t)$和$h_2(t)$十分相近，我们也可以通过取平均值的方法消除病态。数学上可以证明，对于这样的正定矩阵$\Phi_n(z)$，即$G^T(z)$，可以进行如下分解：

$$G^T(z) = F(z^{-1})F^T(z)$$

这时我们只需要将信号通过滤波器$F^{-1}(z^{-1})$，噪声的谱密度矩阵就变为

$$\Phi_w(z) = F^{-1}(z^{-1})\Phi_n(z)F^{-1T}(z) = 2N_0F^{-1}(z^{-1})F(z^{-1})F^T(z)F^{-1T}(z) = 2N_0$$

离散谱矩阵分解过程由手工完成，详见论文[4]，推导和结果见附录二。

4.3维特比解码

x_k经过白化匹配滤波后得到z_k，再通过维特比算法(Viterbi Algorithm)解调得到$\{\hat{a}\}$，该论文选用的分支距离(branch metrics)为

$$\lambda(\hat{a}) = |\sum_{k=0}^{K-1}\sum_{i=0}^{N-1} z_{k,n-i}\hat{y}^*_{k,n-i}| - |\sum_{k=0}^{K-1}\sum_{i=1}^{N-1} z_{k,n-i}\hat{y}^*_{k,n-i}| - \frac{1}{2}\sum_{k=0}^{K-1}|\hat{y}_{k,n}|^2$$

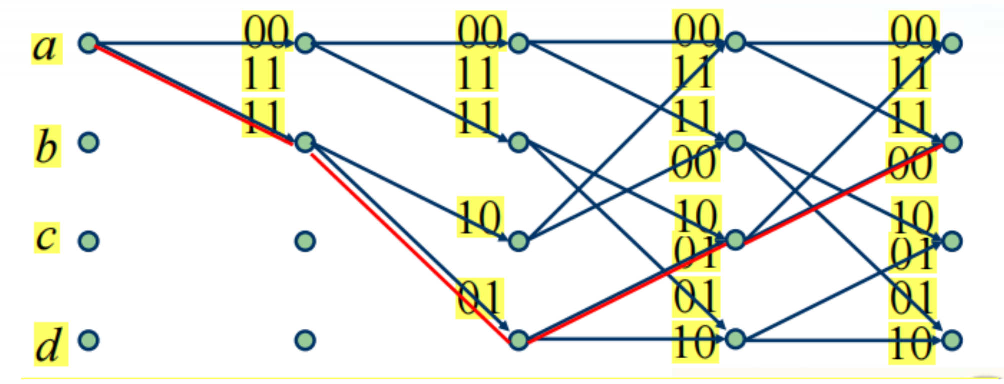
其中

$$y_n = \sum_{l=0}^{L} F_l^T \alpha_{n-l}$$

但是我们注意到，一次分支距离的计算需要N个 {y}，而这N个{y}又需要N+L-1个{α}参与，即对于M元信号一共有$S = M^{N+L-1}$种状态，复杂度太高了，需要进行化简。

对于状态数S（通常情况下是2的整数幂），每个状态代表$depth = log_2(S)$个符号$\{\alpha_n, \ldots, \alpha_{n-depth+1}\}$，剩下的$\{\alpha_{n-depth}, \ldots, \alpha_{n-N-L-2}\}$由到达本状态之前的状态替代，为实现最大似然的效果，选择到达本状态时分支距离最大的状态，进而计算本状态到下一步各状态的分支距离，迭代完成解码过程。

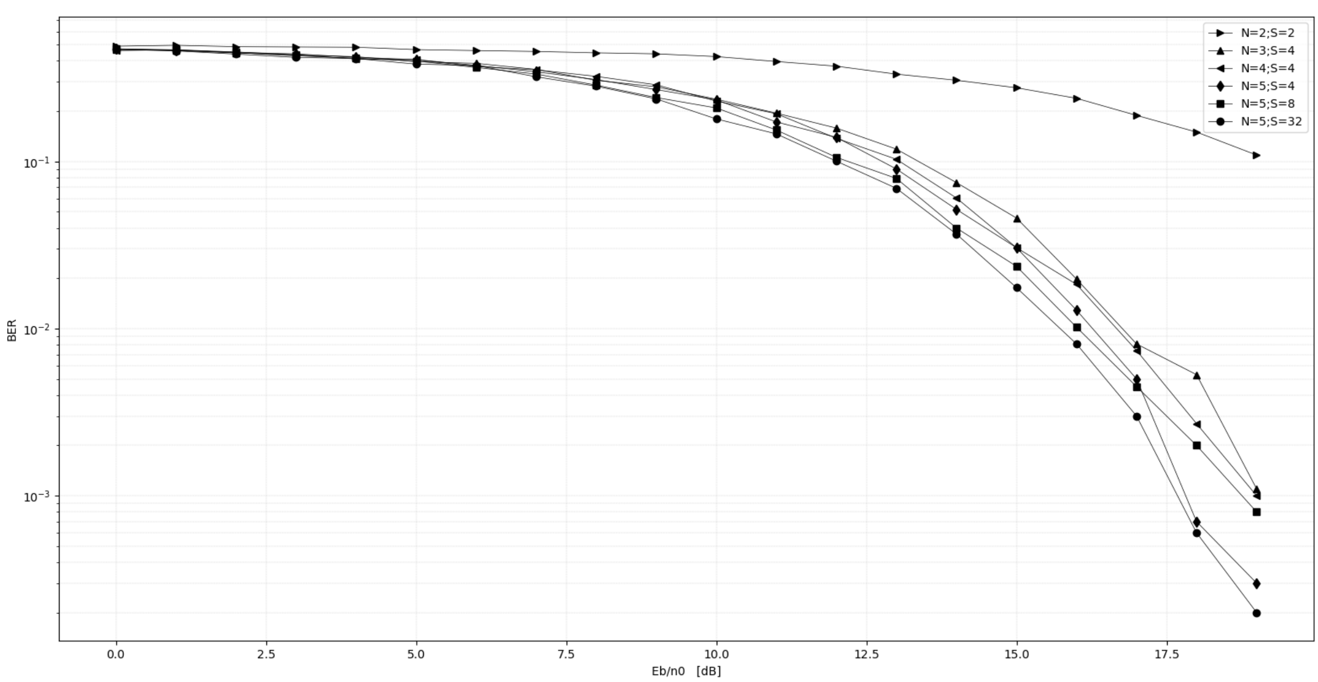
下图可代表$M = 2, S = 4$时的解码过程：



五、实验结果

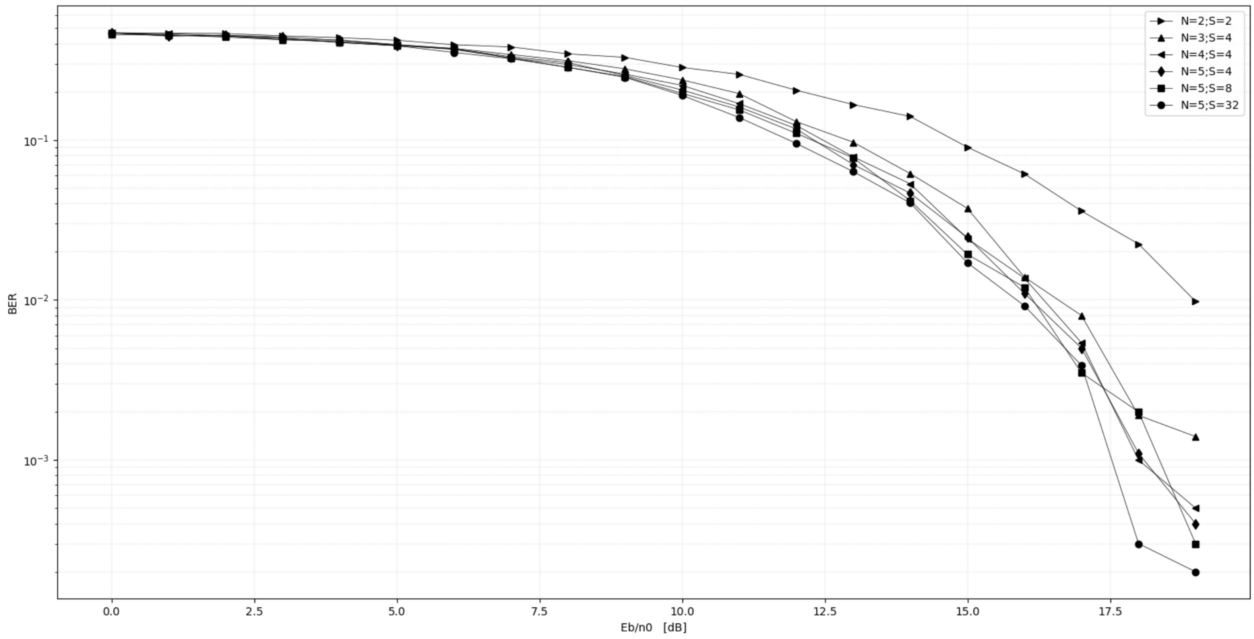
根据经验在$\{a_n\}$序列长度取100000以上时效果较好，但是运行时间太长，故以下图片均为$\{a_n\}$序列长度取10000时测试得到，效果相对会有折扣。

5.1GMSK，K=1



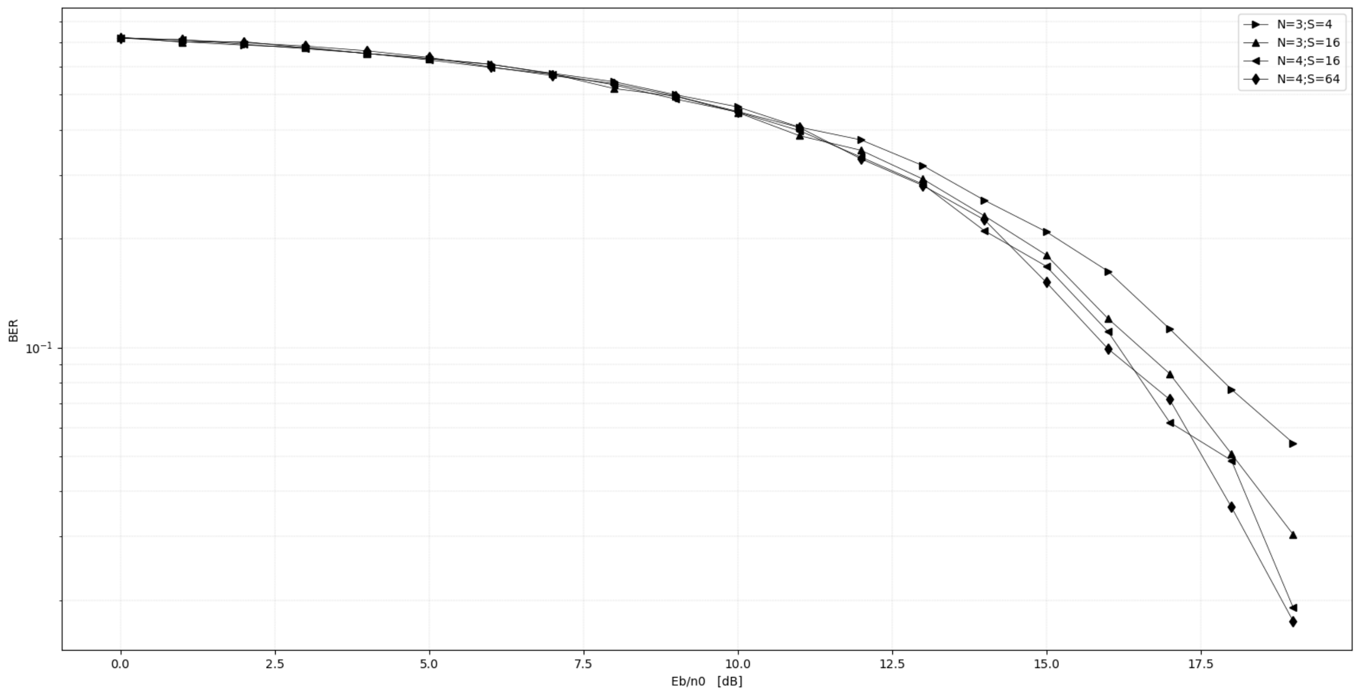
仿真结果与原文相近。BER随SNR的增大而降低；在相同SNR的情况下，BER随S和N的增大而降低。

5.2GMSK，K=2



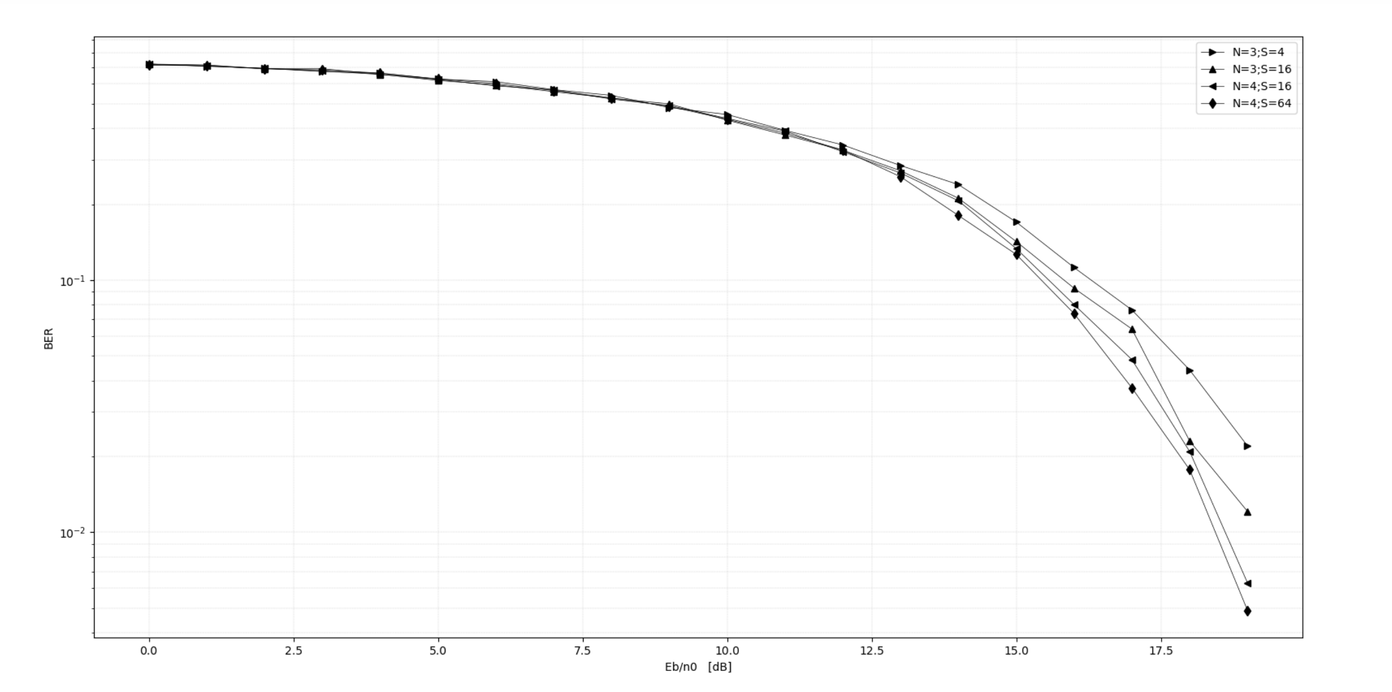
增大K值，发现只有在N=2，S=2得到了明显的改善，其他情况相差不大。推测原因为从报告第二部分可知，GMSK信号的$h_0(t)$幅度是$h_1(t)$的10倍以上，即脉冲的主要信息都在$h_0(t)$上，一般情况下增加K带来的改善不显著；但是N=2，S=2时的状态过于简化，即使增大K带来的微小信息也能有较大改善。

5.3 2RC，K=2



仿真结果与原文相似，但是原文中$k(3,1,1)$和$k=(3,1)$这两条曲线没能理解其意。

5.4 2RC，K=3



增大K值，效果显著变好，可说明$h_2(t)$的引入能够给解调带来较多的信息。

六、代码解读

在本次实验中，我的代码有以下结构：

1.一共有4个文件：

send.py：发送CPM信号

transmit.py：传输信号

receive.py：信号接收与解调

run.py：程序入口，仿真与绘图

程序采用类继承的方式，尽可能复用共有代码。如在发送端，$q_(t)$、$u(t)$、$h_k(t)$、$\beta_{k,i}$等的生成规则是一样的，将这些函数都写在父类中，对于不同的CPM信号，我们只需要在子类中覆写频率脉冲函数$g(t)$即可完成；又如对于不同元(M)的信号，$h(t)$的生成规则是不同的，但实际上2元可以看做是多元的简化版本，因此实际上一个函数即可实现，而不需要分情况讨论。

2.为提高仿真速度，程序中所有连续信号均被离散信号近似替代，积分采用梯形公式，采用Simpson公式可以进一步提高精度，但梯形公式已可满足本次实验。所有连续函数在send.py文末的注释中，时间充裕的话使用连续函数仿真应该会得到更精准的结果。

3.为提高仿真速度，程序中使用了较多的高维数组运算，很多地方“酷似”MATLAB风格，如$u(t)$函数以$t = Lg$为对称轴，故我们只需要计算一半，再用一行代码即可翻折过去：

```
            u[l][len(q)-1::1] = u[l][len(q)-1::-1]
```

4.程序的可扩展性较强。如论文中用的高斯信道，我们可以添加以外的信道；很多函数是可以通用的，而不局限于GMSK和2RC信号；对于暂时没开发的部分，如果调用会发生raise NotImplementedError报错。

5.程序的自动化程度还是不够高。如G(z)矩阵的分解暂时还是用手工来计算的。

七、总结

对我来说这是一次工作量比较大的复现，主要是因为相关知识储备较少，所以在工作前期进展非常缓慢，大部分时间都用在了查阅文献上，比如光GMSK的表达式我就查了一天，因为我竟然查到了好几个版本……渐渐入门之后工作就快起来了，但我总是不满意代码而推翻重写，我不想写一个只适用这篇论文的代码，我想写出的是，随便给我一个新的CPM信号，我5分钟就能把它跑出来的那种代码，最终也算有所小成，毕竟我只需要覆写一下$g(t)$函数，可能5分钟都用不上。

总的来说本次实验锻炼了我从0(约等于)到1的过程，我感觉自己写代码的能力也得到了提升。

八、参考文献

[1]https://en.wikipedia.org/wiki/Continuous_phase_modulation

[2]https://www.southampton.ac.uk/~sqc/EL334/Coh-non.pdf

[3]Decomposition of Mary CPM Signals into PAM Waveforms

[4]The factorization of discrete- process spectral matrices

附录一：多元CPM信号分解

相比于2元的情况，多元CPM信号的分解比较复杂，首先我们根据M得到P

$$2^{P-1} < M \leq 2^P$$

对于$0 \leq l \leq P-1$，有

$$u^{(l)}(t) = \begin{cases} sin[2\pi h^{(l)}q(t)]/sin(h^{(l)}\pi), & 0 \leq t \leq LgT \\ u^{(l)}(2LgT - t), & LgT < t \leq 2LgT \\ 0, & otherwisw \end{cases}$$

$$h_k^{(l)}(t) = \prod_{i=0}^{Lg-1} u^{(l)}(t + iT + \beta_{k,i}LT) \qquad 0 \leq k \leq Q-1$$

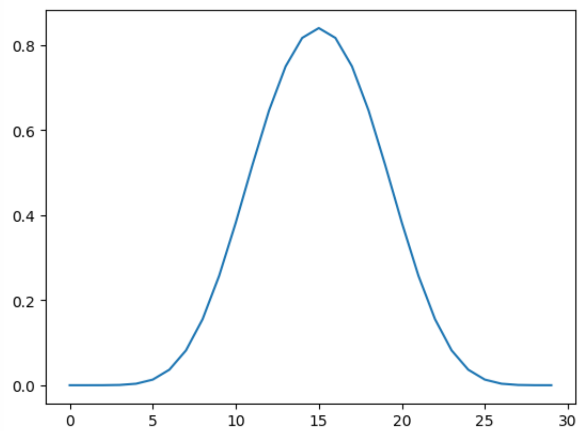
$$
b_{n,k}^{(l)} = exp\{jh^{(l)}\pi[\sum_{m=-\infty}^{n} a_m - \sum_{i=0}^{Lg-1} a_{n-i}\beta_{k,i}]\}
$$
$$
\alpha_{0,n} = b_{0,n}^0 b_{0,n}^1
$$
$$
\alpha_{1,n} = b_{1,n}^0 b_{1,n}^1
$$
$$
\alpha_{2,n} = b_{2,n}^0 b_{2,n}^1
$$
$$
\cdots\cdots
$$

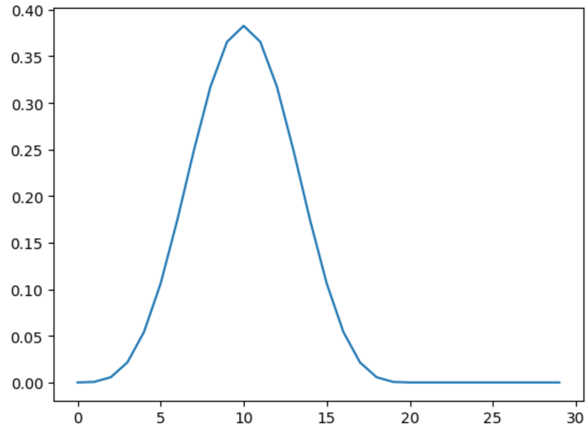
此时得到的$h_k^{(l)}(t)$并不是最终的脉冲函数，还需要进行一系列规则的对应乘加，对2RC信号，即M=4时，论文[3]给出了如下对应表格：

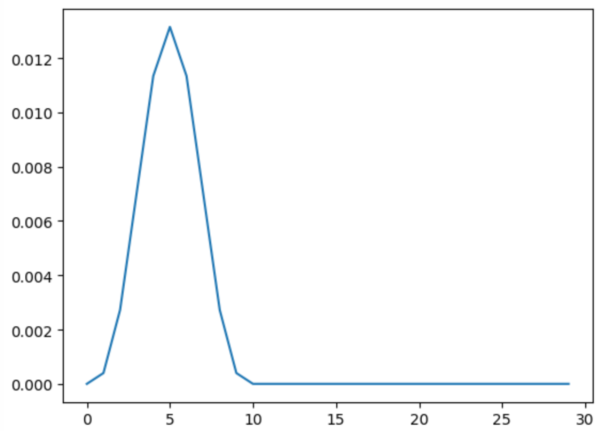
TABLE V
LAURENT FUNCTIONS FOR $M = 4$, $L = 2$

k	j	m	$\mathbf{e}_j^{(m)}$		$g_k(t)$	$a_{k,n}$
0		0	0	0	$c_0^{(0)}(t)\,c_0^{(1)}(t)$	$b_{0,n}^{(0)}\, b_{0,n}^{(1)}$
1		1	0	1	$c_0^{(0)}(t+T)\,c_0^{(1)}(t)$	$b_{0,n-1}^{(0)}\, b_{0,n}^{(1)}$
2	0	2	1	0	$c_0^{(0)}(t)\,c_0^{(1)}(t+T)$	$b_{0,n}^{(0)}\, b_{0,n-1}^{(1)}$
3		3	0	2	$c_0^{(0)}(t+2T)\,c_0^{(1)}(t)$	$b_{0,n-2}^{(0)}\, b_{0,n}^{(1)}$
4		4	2	0	$c_0^{(0)}(t)\,c_0^{(1)}(t+2T)$	$b_{0,n}^{(0)}\, b_{0,n-2}^{(1)}$
5		0	0	0	$c_1^{(0)}(t)\,c_0^{(1)}(t)$	$b_{1,n}^{(0)}\, b_{0,n}^{(1)}$
6	1	1	1	0	$c_1^{(0)}(t)\,c_0^{(1)}(t+T)$	$b_{1,n}^{(0)}\, b_{0,n-1}^{(1)}$
7		2	2	0	$c_1^{(0)}(t)\,c_0^{(1)}(t+2T)$	$b_{1,n}^{(0)}\, b_{0,n-2}^{(1)}$
8		0	0	0	$c_0^{(0)}(t)\,c_1^{(1)}(t)$	$b_{0,n}^{(0)}\, b_{1,n}^{(1)}$
9	2	1	0	1	$c_0^{(0)}(t+T)\,c_1^{(1)}(t)$	$b_{0,n-1}^{(0)}\, b_{1,n}^{(1)}$
10		2	0	2	$c_0^{(0)}(t+2T)\,c_1^{(1)}(t)$	$b_{0,n-2}^{(0)}\, b_{1,n}^{(1)}$
11	3	0	0	0	$c_1^{(0)}(t)\,c_1^{(1)}(t)$	$b_{1,n}^{(0)}\, b_{1,n}^{(1)}$

K=3时，$g_k(t), k=0,1,2$脉冲序列波形分别如下：







附录二、离散谱矩阵分解

以比较简单的GMSK、K=1的情况为例，$G^T(z)=G(z)=g_{-2}z^{-2}+g_{-1}z^{-1}+g_0z^0+g_1z^1+g_2z^2$，其中$g_{-2}=g_2,g_{-1}=g_1$，设其零点为$z_i,i=0,1,2,3,$，且$|z_0|<\ldots<|z_1|$，则有$F=\frac{(1-z_0)(1-z_1)}{\sqrt{\sum g_i}([1,-z_0]\otimes[1,-z_1])}$

GMSK、K2时，有

$$F(z)=\begin{bmatrix}0.08024+0.6558z^{-1}+0.4442z^{-2} & 0.2305+0.45977z^{-1}\\ 0.0263 & 0.0421\end{bmatrix}$$

2RC、K2时，有

$$F(z)=\begin{bmatrix}0.1187+0.6935z^{-1} & -0.0935\\ 0.01723+0.1282z^{-1} & -0.11299+0.1558z^{-1}\end{bmatrix}$$

2RC、K=3时，有

$$F(z)=\begin{bmatrix}0.1532+0.6015z^{-1} & -0.0302+0.2923z^{-1} & -0.1279\\ 0.2044+0.0379z^{-1} & -0.019193+0.1778z^{-1} & -0.07903\\ 0.0053 & 0.0028 & 0.0043\end{bmatrix}$$

附录三、实验参数设置

GMSK:

```
BT = 0.25
h = 0.5
T = 1
Es = 1
Lg = 2
fs = 10
BT = 0.25
M = 2
L = 2
```

RC:

```
BT = 0.25
h = 0.25
T = 1
Es = 2
Lg = 2
fs = 10
M = 4
L = 1
```