通信原理Matlab第三次仿真作业——模拟信号数字化

# 题目要求

用计算机仿真模拟信号数字化系统过程。量化器为均匀量化，量化范围V；输入信号假设服从均匀分布。二进制信道传输误比特率。

**分别仿真得到以下曲线，并将它们绘制在一张图中**：

1）量化比特（含符号位）、采用自然码编码时，作出输入信号平均功率从-60dBW至0dBW（电阻1欧）间变化时输出信噪比变化曲线；

2）量化比特（含符号位）、采用折叠码编码时，作出输入信号平均功率从-60dBW至0dBW（电阻1欧）间变化时输出信噪比变化曲线；

3）量化比特（含符号位）、采用自然码编码时，作出输入信号平均功率从-60dBW至0dBW（电阻1欧）间变化时输出信噪比变化曲线；

4）量化比特（含符号位）、采用折叠码编码时，作出输入信号平均功率从-60dBW至0dBW（电阻1欧）间变化时输出信噪比变化曲线；。

关键：如何统计输出信噪比，关键是统计平均输出噪声功率

按照输入信号平均功率生成多个（如10000）个样值，对每一个样值作为输入信号，进行量化、编码、信道传输（产生随机误码）和译码，将恢复出的样值与输入样值作差取平方得到样值误差功率。最后，取平均得到平均输出噪声功率。

# 仿真实现过程

## 自然码和折叠码编码的实现

实际上自然码就是将对应量化区间的量化级数转为二进制的表示。因此只需要利用Matlab根据量化级数计算公式计算出输入信号对应的量化级数，再利用Matlab内部的de2bi()函数将量化级数转为二进制即可。

折叠码编码只需在自然码编码的基础上将最高位为0的编码的低几位取反即可得到自然码对应的折叠码。为此可利用Matlab中对矩阵元素的逻辑运算find()函数，即将最高位取反，再利用find()函数找到最高位不为0的行数，再将对应行数的低N-1位取反即可。

## 随机误比特的产生

仿真题目要求误比特率为1e-5，即在将输入信号的编码再通过信道之后，在每一个二进制位与编码后的信号位数相反的概率为1e-5。为此考虑了两种实现方法，并且都加以实现，得到了相似的结果。

产生误比特的第一种方法即按照基本的要求，产生一组0到1的与编码后的信号同维度的随机数矩阵Num\_detect，利用矩阵的逻辑运算1.\*(Num\_detect < 1e-5) + 0.\*(Num\_detect >= 1e-5)，即可得到小于1e-5的元素对应的位置，再将对应的编码后的信号矩阵的对应元素取反即可。

另一种方法是将编码后的二进制信号直接利用reshape()函数重组成一列得到总数据量size，再利用随机数产生size\*Pe个0到size的随机整数即可。

两种产生误比特的方法对比而言最终效果类似，利用的思想不同。其中第一种方法更接近于实际的概率情况，第二种方法则利用合理的理论推导进行了近似，操作起来更为简单，运行速度更快。

## 2.3 超出量化范围时信号的压缩

当输入信号功率大于-4dBW时，输入信号的幅度超出了量化器给定的[-1,1]V的范围，为此考虑将幅度大于等于1和小于等于-1的信号的幅度都压缩为0.9999（符号保持不变），而保持其它信号不变。再将压缩后的信号进行量化具体的操作同样利用了分段函数的思想，与随机数产生误比特的方法类似。

## 2.4 仿真代码的优越性

由于Matlab以矩阵运算为优势，对循环语句的执行效率很低。为了提高代码的运行速度，减小计算机的运行负担。使得能够计算更多的样值，获得更准确的结果，在编写代码的时候所有的语句都采用了矩阵运算，充分利用了Matlab的优势。

具体到仿真实现上，将样值点数取到了1e6，最终运行一次的时间可以维持在2分钟以内。

# 仿真结果及简单解释

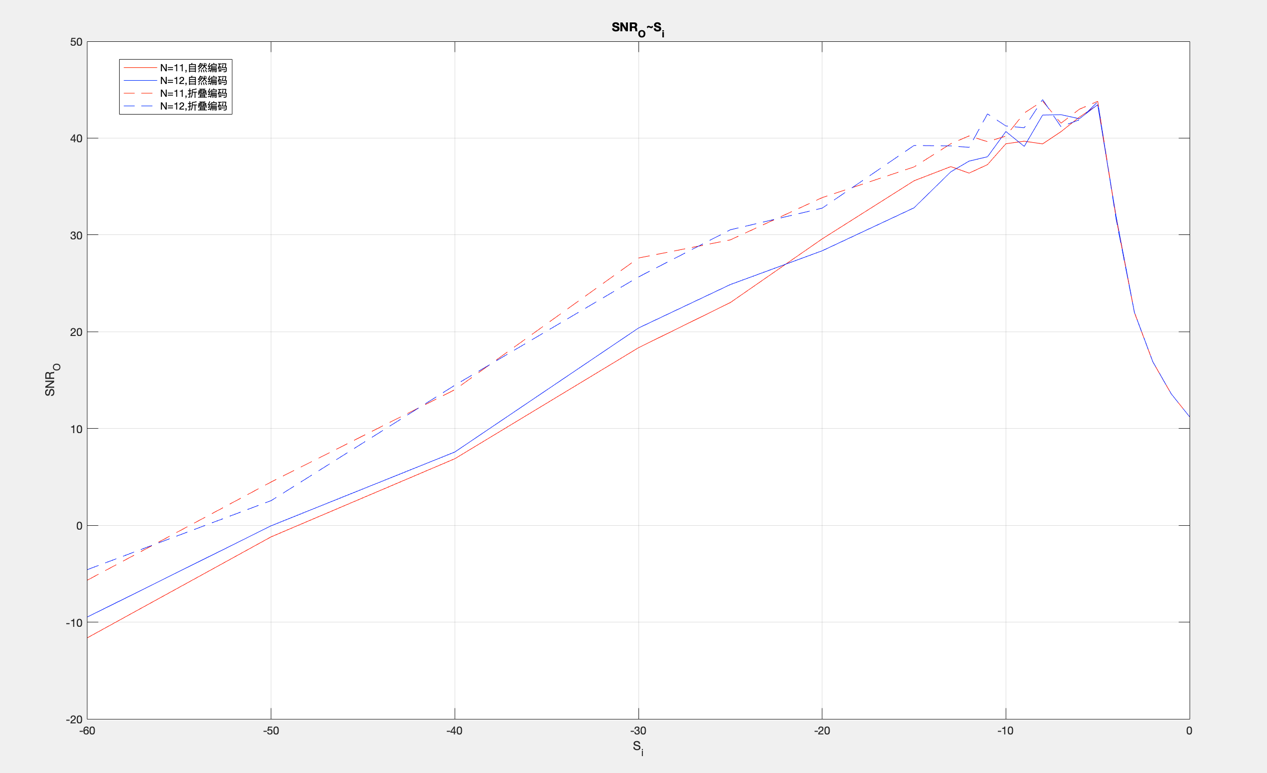


图 1 输出信号信噪比与输入信号功率关系曲线一

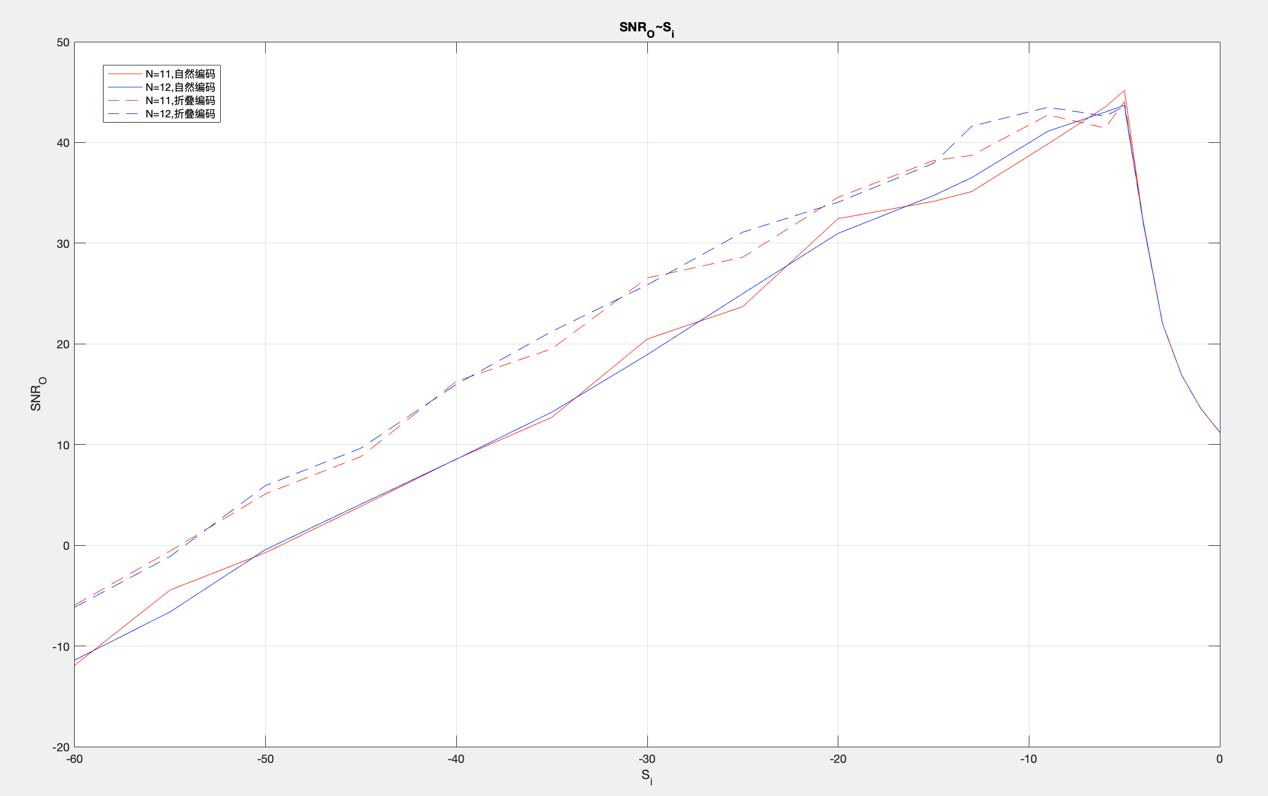


图 2 输出信号信噪比与输入信号功率关系曲线二

以上两幅图为输入功率取不同的点得到的输出信噪比的结果（样值点数为1e6）。其中蓝色曲线代表12位量化位数，红色曲线代表11位量化位数。虚线代表折叠编码，实线带阿标自然编码。其中图一图二均为不等间隔取点（图二在输入信号功率更低时取点更密），其中越接近0取点越密。

通过两幅图的对比可以看出，在对于同样的编码方式，量化位数为12位时信噪比总体高于量化位数为11位时。而对于同样的量化位数，输入信号功率小的时候折叠码的信噪比明显高于自然码的信噪比，随着输入功率逐渐增大两者的输出信噪比逐渐接近。当信号超出量化范围时，信噪比急剧降低，不同量化位数和编码方式对信噪比的效果几乎相同。

经过对两种编码方式的仿真实验，得到了与理论一致的结果，也让我对两种编码方式的本质以及其特点有了更为深刻的认识。