**编码引论第二次大作业**

**编码调制及安全编码**

**实验报告**

姓 名：张鹤龄

班 级：无04

学 号：2020010735

完成时间：2022/11/18

目录

[一、编码调制 3](#_Toc120221491)

[一、调制和解调 3](#_Toc120221492)

[二、性能讨论 3](#_Toc120221493)

[三、绘制信号波形和功率谱 3](#_Toc120221494)

[四、不同比特映射、收尾、判决方式下误比特率和误字率与E\_b/n\_0 的关系曲线: 3](#_Toc120221495)

[五、尝试不同的K\T组合，特别是当他们不是整数倍关系时，看结果有什么不同: 3](#_Toc120221496)

[二、安全编码 3](#_Toc120221497)

[一、编码思路 3](#_Toc120221498)

[二、RSA加密算法 3](#_Toc120221499)

[加密原理 3](#_Toc120221500)

[算法及实现 3](#_Toc120221501)

[安全性和性能 3](#_Toc120221502)

[三、AES(Rijndael)加密算法 3](#_Toc120221503)

[加密原理 3](#_Toc120221504)

[算法及实现 3](#_Toc120221505)

[安全性与性能 3](#_Toc120221506)

[四、DES加密算法 3](#_Toc120221507)

[算法及实现 3](#_Toc120221508)

[安全性与性能 3](#_Toc120221509)

[五、实验验证 3](#_Toc120221510)

[三、附录 3](#_Toc120221511)

[成员分工： 3](#_Toc120221512)

[文件清单： 3](#_Toc120221513)

# 一、编码调制

## 一、调制和解调

对于第i个需要传输的符号，信源发送标准复电平，其中。之后对乘上旋转因子，得调制后的离散信号, 其中。可见，调制过程采用调相方式将信息调制到不同初相的正弦波上，并以Ts为采样周期（Ts不必有特定值，只需收发端约定好即可）在此正弦波上均匀采T个点。

调制后的信号在信道中传输的过程中会叠加噪声：

收端对收到的信号乘上反向旋转因子解调，

进行加和平均后得到解调结果：

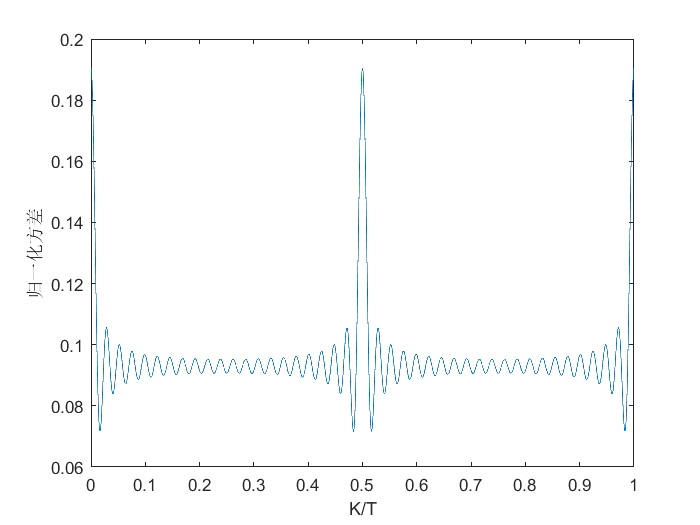
第一项是原信号，第二项是固定偏差。由于偏差是固定的，可以做相对的优化，在此只列出一个方案：

记，在发端，规定标准电平为，收端的标准电平不变。这样通过调制和解调后，收端收到的就是标准电平叠加上一定的噪声。至于如何寻找，可以尝试求解析解，也可以通过梯度下降、切比雪夫函数等方法拟合，这里不必给出具体的解决方案。

## 二、性能讨论

对于vi中包含的随机误差，计算均值和方差：

除以以归一化，在Matlab中绘制归一化方差和K/T的关系，结果如下：

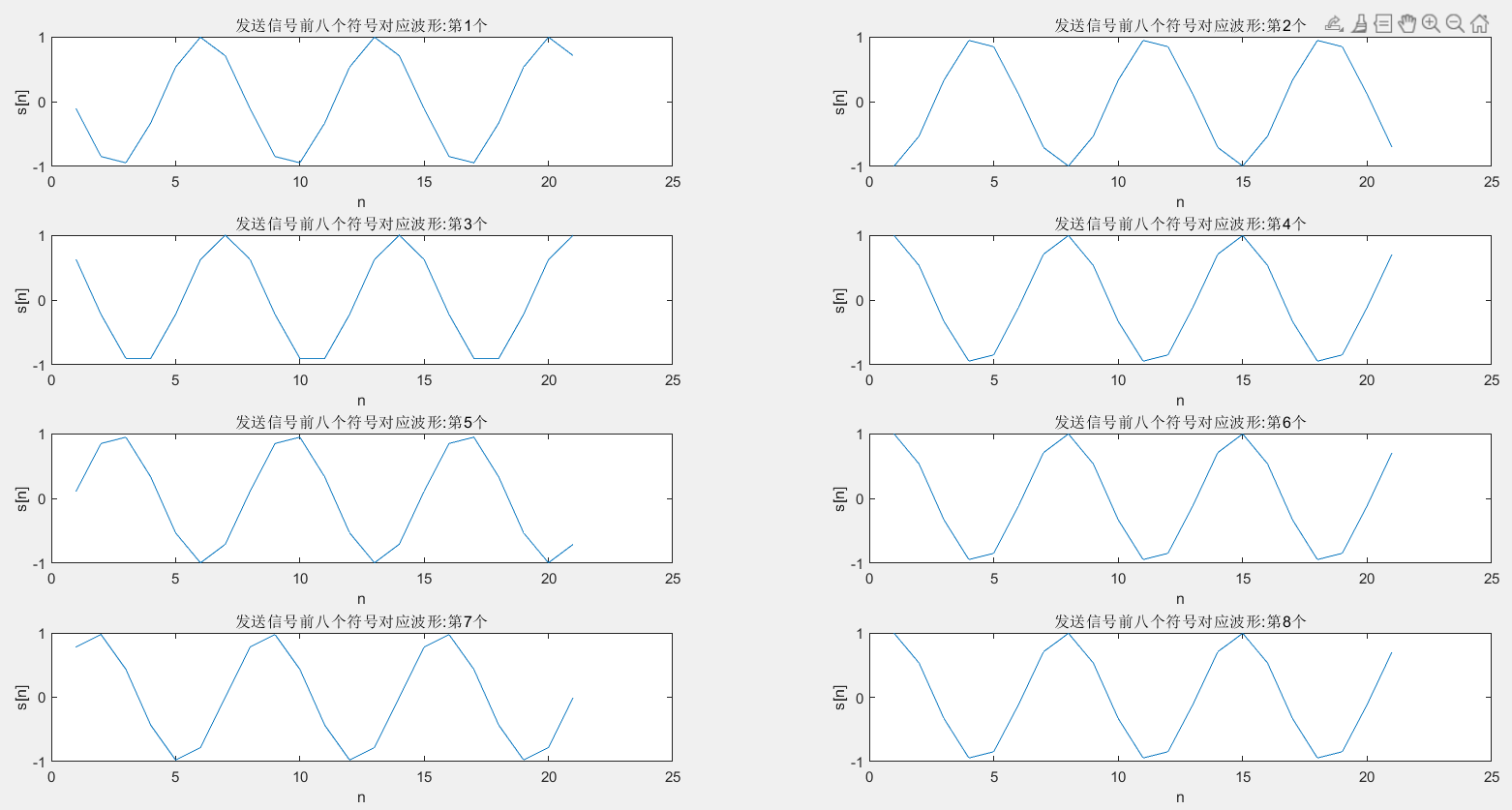


上图中取T=21，由于取了平均的原因，噪声方差较未采样的方差小了很多。进一步，为了取得更好的效果，可以选择合适的K，使得K/T恰好为上图中的最小值点，以最小化方差。

## 三、绘制信号波形和功率谱

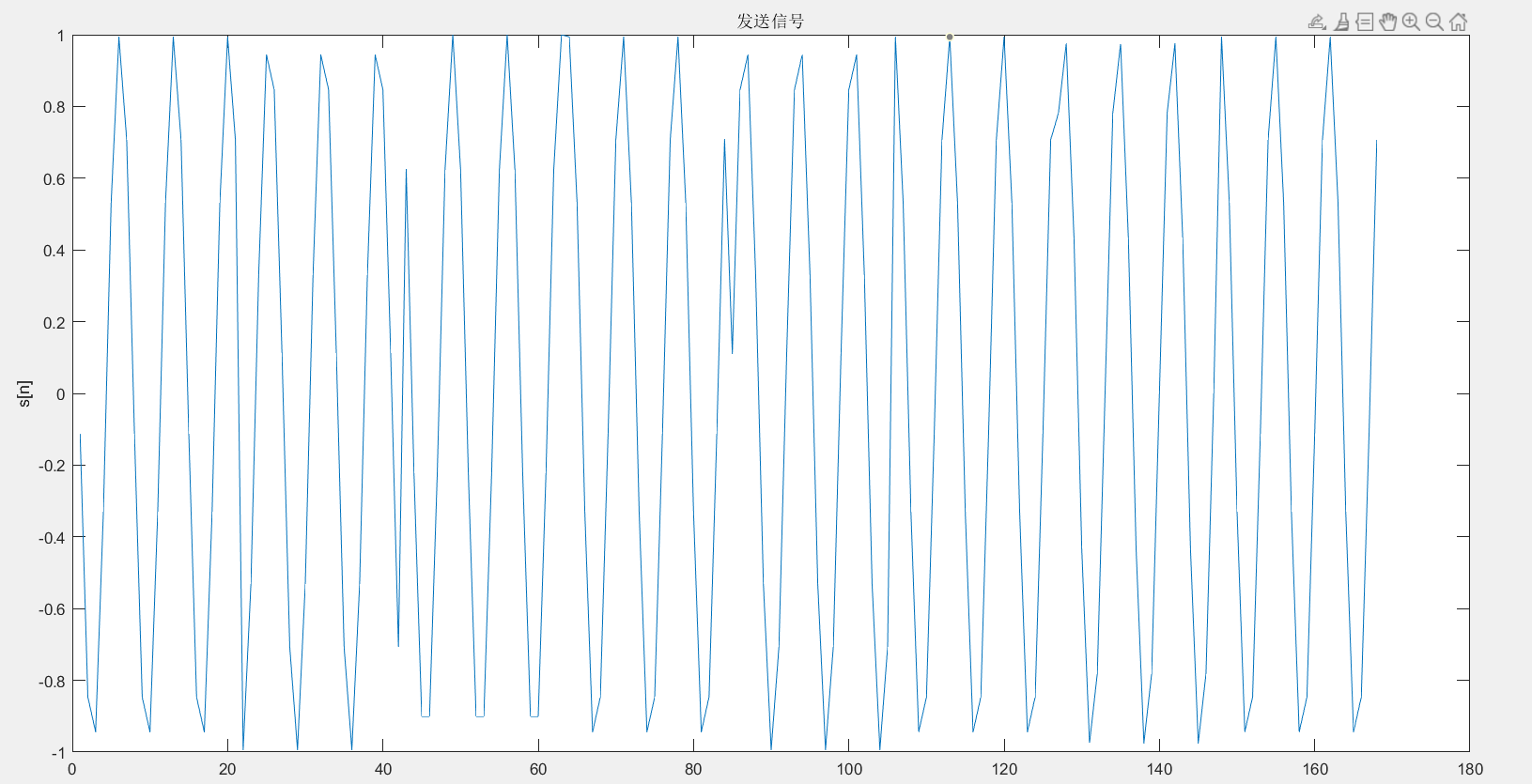
在K=3，T=21的情况下：

* 绘制每个符号的标准波形：



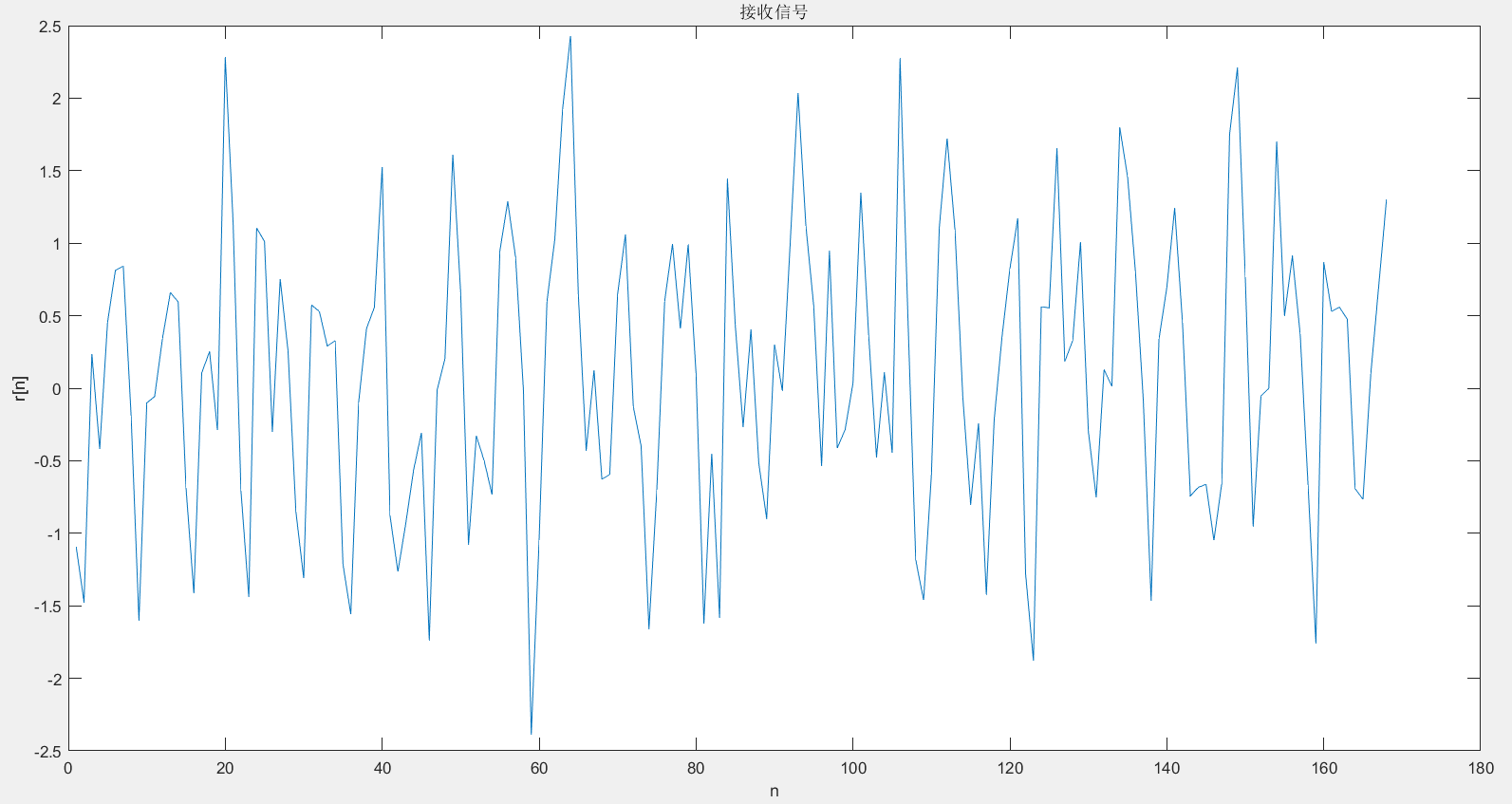
由于采取调相方式，因此不同符号的信号波形为相同频率不同初相的正弦波。

* 发送信号的波形：



发送信号的波形整体呈现正弦波，但中间由于发送符号的切换存在相位突变。

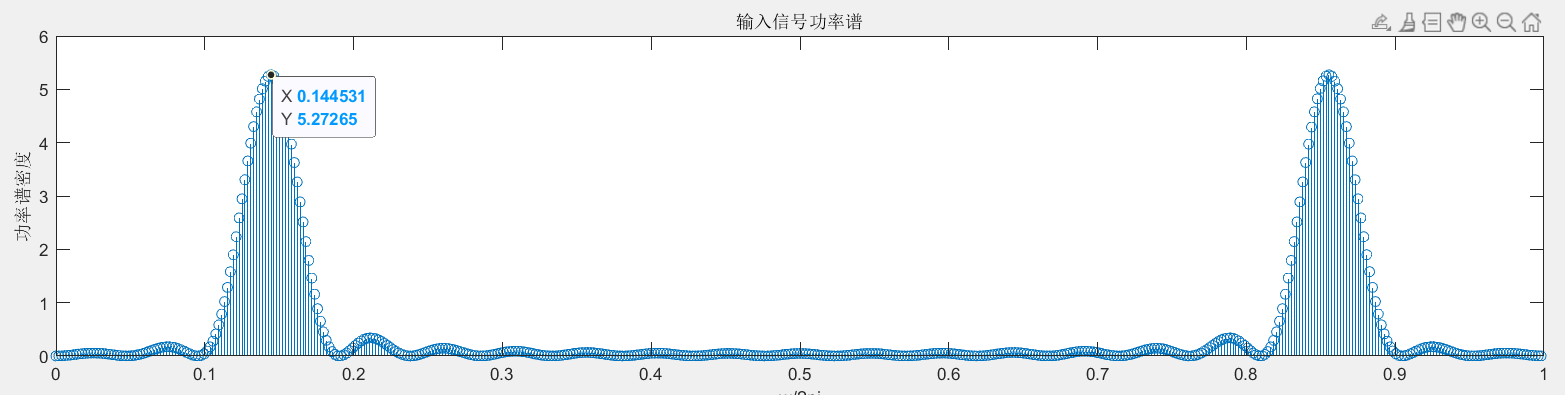
* 接收信号的波形：



由于叠加了噪声，接收信号幅度和相位都出现了一定程度的起伏。

* 发送信号的功率谱：

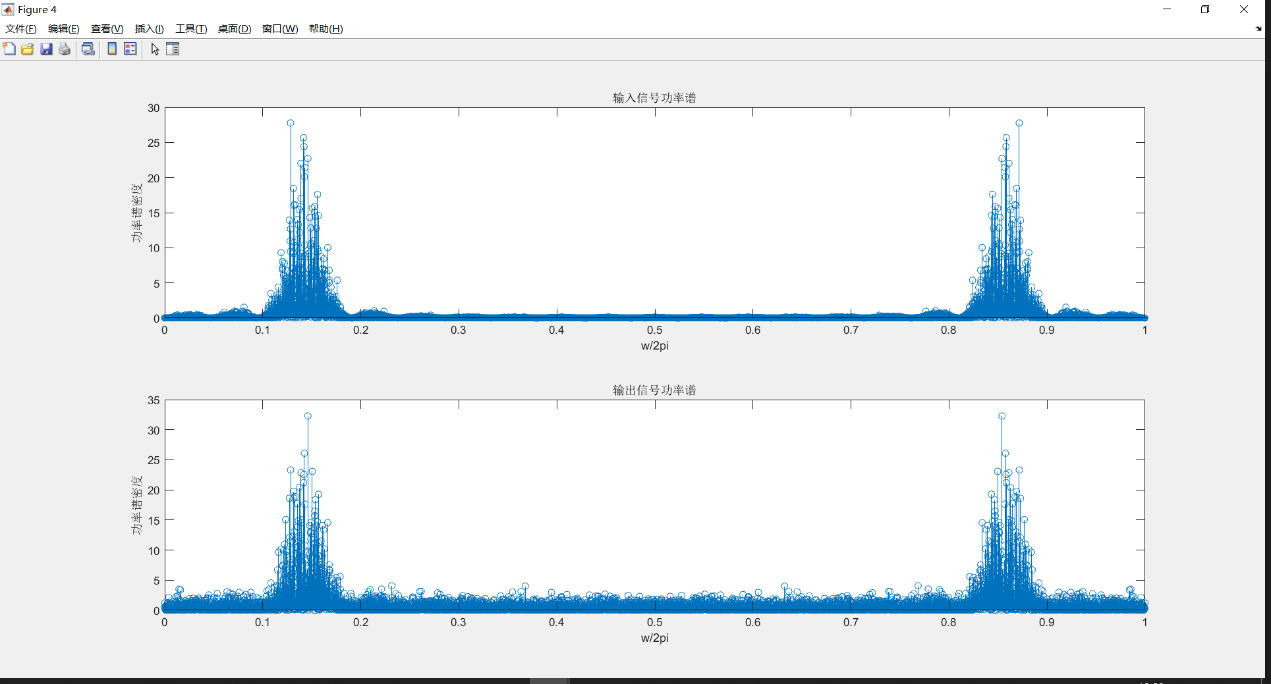
单个信号的功率谱如下：



功率谱密度为Sa2, 中心频率为。

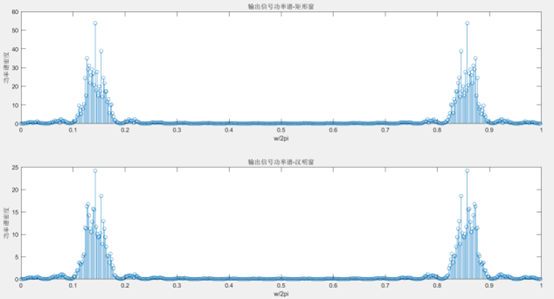
（如果做长度为T的FFT，则只在中心频点有值。这是因为，此时时域采样点数M等于变换点数N，对于单个谱分析滤波器，在非主峰的其他采样点取值均为0（分子sin(M/2\*θ)，θ从0开始，采样间隔为2π/N，故分子除主峰外恒为0））

* 对整体信号做傅里叶变换：



中心频点仍然在，但由于相位跳变，一些杂波分量被引入。

另一种分析方法是进行STFT，因为各时间段的频谱性质大致相同。以下是STFT分析结果：



取一段较短时间的信号分析，结果和取所有信号分析频谱效果几乎相同。这样的频谱分析方法利用了各时间段的频谱性质大致相同的特点，大大提高了频谱分析的即时性。

## 四、不同比特映射、收尾、判决方式下误比特率和误字率与E\_b/n\_0 的关系曲线:

不同收尾方式下，误比特率、误字率没有明显的差别：

图表, 折线图

描述已自动生成图表, 折线图

描述已自动生成

不同比特映射方式下，2bit误字率和误比特率略微高于3bit；

图表, 直方图

描述已自动生成图表, 直方图

描述已自动生成

由于2bit电平间距较大，因此差错概率也较低。

不同判决方式下，软判显著好于硬判；

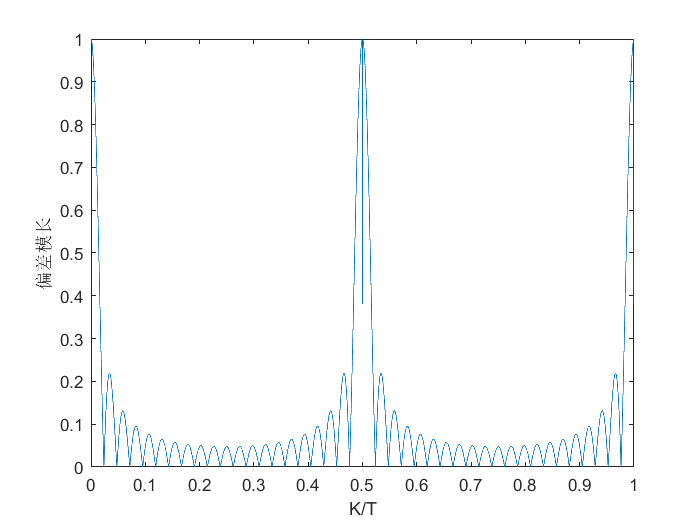
图表, 直方图

描述已自动生成图表

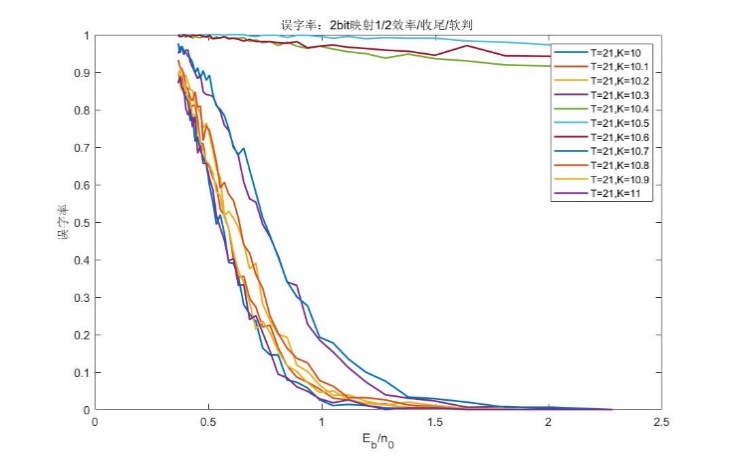
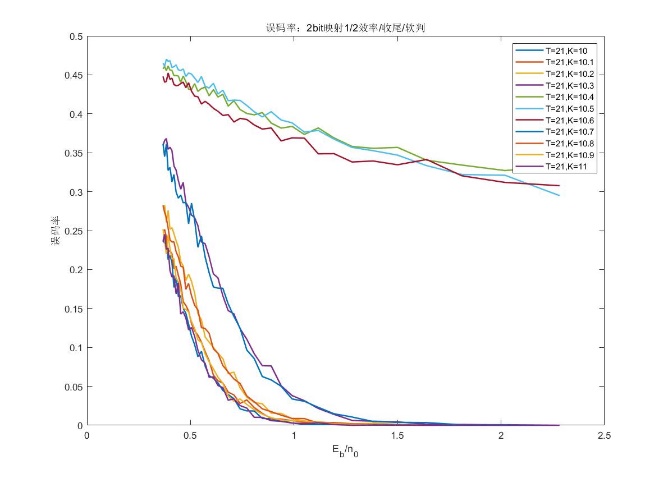
描述已自动生成

## 五、尝试不同的K\T组合，特别是当他们不是整数倍关系时，看结果有什么不同:

这里考虑固定偏差（即解调后得到vi表达式中的第二项）。画出其随K/T变化的曲线，如下：



与随机偏差的方差对比，可以看到固定偏差是偏差的主要来源。为测定其对信道性能的影响，在2bit映射/收尾/软判方式下，测定不同K/T下的误码率：



仿真结果完全印证了理论分析。K/T=0.5时，误字率和误码率均为最大值，这对应固定偏差的峰值；10.4/10.6次之，10.3/10.7又次之且误码率陡降，这反映了固定偏差的对称性以及对称轴附近的剧烈变化。

# 二、安全编码

## 一、编码思路

加密分为对称加密和非对称加密，两种方法各有其优缺点。对称加密的双方共享相同密钥，加密速度快，但需要分发者生成密钥并通过保密信道传输密钥，密钥可能被直接获取；非对称加密的加密密钥和解密密钥不同，不需要分发者，从而避免了其带来的不安全性，但同时加密速率也较低，不适合加密过长的信息。因此，在实际传输任务中，经常采用对称加密的方式加密要传输的信息，而用非对称加密的方式加密对称加密的密钥。本实验就采用这种方式，采用对称加密法（AES/DES）加密信息，同时用非对称加密法（RSA）加密其密钥，以实现安全传输。

## 二、RSA加密算法

### 加密原理

RSA算法利用超长整数质因数分解的极大复杂度保证非对称加密的安全性。以下为其算法流程：

定义n的欧拉函数为

其中pi为n的第i个质因数。由欧拉公式，

由n可以自然地计算Φ(n)。为增大对n质因数分解的难度，n应当为两个大质数的乘积，即n = p \* q, Φ(n) = (p – 1)(q – 1)。

利用Φ(n)可以生成公钥Pu和私钥Pr。在加密时，计算明文的Pu次幂；在解密时，计算密文的Pr次幂，还原出明文。故应当有

与欧拉公式对照便可以选满足的公钥和私钥。将这个式子稍作调整，得到

即，公钥可以选择与Φ(n)互素的数，而私钥为公钥与Φ(n)的裴蜀系数，可以用扩展欧几里得算法计算出来。

上述过程在私钥的持有者处完成。生成三元组Pu, Pr, n后，私钥持有者将Pu和n公布，自己保留Pr而将中间变量全部销毁。这样，由于对n进行质因数分解非常困难，外部无从根据Pu和n计算出Pr，也就无法还原出明文，从而保证了加密的安全性。

### 算法及实现

RSA加密中用到的主要算法如下：

寻找大素数：Miller-Rabin算法；

模n幂运算：Montgomery算法；

解裴蜀系数：扩展欧几里得算法。

下面对这些算法的原理和具体实现作简要说明。

#### Miller-Rabin算法

Miller-Rabin算法的实质是加入二次探测的费马小定理检测。由费马小定理，若p为素数，则

但这仅仅是p是素数的必要条件，而非充分条件。我们需要用其他方法试验p，进一步提升其为素数的概率。

引入二次探测定理：若

则由唯一分解定理，a-1和a+1中一定有一个是p的倍数，即

但若p不是质数，上式就不一定成立，因为p的质因数可能分别存在于a-1和a+1中。回到费马小定理检测，如果p≠2，且p是素数，那么p-1一定可以表示为2kd的形式，即

对上式应用二次探测，得到p是质数的必要条件为

用多个a对p重复进行实验，若p全部通过，则有相当大的把握认为p是一个素数。

在我们的实验里，相关部分被封装在findPrime函数中。由于我们采用了1024位RSA算法，因此需要生成2个512位素数，为此需要先生成512位随机数再对其进行素性检验。为保证效率，若生成的随机数为偶数，就将其加一再进行检测；之后测试其能否整除10000以内的素数，排除大多数合数；最后挑选20个a进行测试，若p全部通过，则p为素数的概率约为1-2-40，可以认为p是素数。为加快检验速度，a从10000以内的素数中随机产生。

#### Montgomery算法(快速幂)

Montgomery算法用于计算大指数幂运算。具体算法如下：

要计算ak，先将k分解为

则

ci=0或1。而可以由平方计算而来，从而由不断平方可以计算出每个，从而计算出ak。

在我们的实验里，相关部分被封装在montMod函数中。首先将指数表示为其二进制形式，递推计算并依据指数二进制形式中的对应位决定是否将当前的乘入最后结果中，如此计算出ak。

#### 扩展欧几里得算法

扩展欧几里得算法用于计算公钥对应的私钥。具体算法如下：

欧几里得定理保证了对于互素的两个数a, b存在整数x, y，满足

其中x, y是裴蜀系数。上式还可以写成

x是a在模b域中的逆元，因此可以通过计算裴蜀系数的方式计算出某元素在模域中的逆元。

我们可以通过递归的方式找出x和y。从gcd(a, b)=1, 由欧几里得算法可以推出gcd(b, a – b \* int(a/b))=1，因此也存在x’和y’满足

上式与

联立，得出x, y和x’, y’的递推关系：

这样我们就把求解x, y转换成了求解x’, y’，也即b, a – b \* int(a/b)的裴蜀系数。由于后一个问题规模较小，不断递归求解就可以得到x, y的值。递归终点为当a = 1，b = 0时，裴蜀系数为1和0。

求出x，y后，y是b在模a域中的逆元。将Pu代入b，Φ(n)代入a，求出的y便可作为Pr。本次实验的扩展欧几里得算法封装在exEuclid函数中。[[1]](#footnote-1)

#### RSA加密

利用上面的算法可以进行RSA加密。首先用findPrime函数找到三个512位素数，其中两个作为生成n的p和q，另外一个作为公钥。之后，利用公钥Φ(n)计算出私钥。加密模块将AES密钥转化为128位大整数a，并用montMod计算aPu；解密模块则用montMod计算aPuPr，还原出大整数a，将其转化为明文。

### 安全性和性能

本次实验实现的RSA算法为1024位，足够对AES密钥（128）位和DES密钥（64位）进行加密，因而总密钥数为2128；目前可以暴力分解的整数最高为768位，故1024位RSA足够保证安全性[[2]](#footnote-2)。由于RSA仅用于加密少量信息，因此对其性能的要求不必过于严格。

## 三、AES(Rijndael)加密算法

### 加密原理

对于对称加密来说，攻击方可能已经掌握了一定量的密文，或一定量的明文以及对应的密文，甚至可能是任意明文以及对应的密文，而其目标便是通过掌握的信息获取密钥，从而对之后获得的密文进行解密（或冒充发信者对明文进行加密）。因此，如何尽可能隐藏密钥是对称加密面临的最重要问题。AES加密算法通过重复置换-代换操作，在使得加密具有强非线性（非差分可破）的同时，让密文中的每个字符和明文中的每个字符有关，带来了显著的雪崩效应，进一步加强了加密安全性。

### 算法及实现

AES由密钥加法层、字节代换层、行位移层和列混淆层组成。

#### 密钥加法层

密钥加法是AES算法中唯一具有加密特性的部分，除此之外，AES的置换规则、代换表全部公开，无加密特性可言，它们只起到加强非线性和雪崩特性的作用。

AES采用分组加密的方式，每次能加密的明文长度等同于密钥长度。本次实验采用128位密钥，故每次加密的明文也为128位，恰为16个字符。这样，明文就可以排成如下的矩阵形式：

|  |  |  |  |
| --- | --- | --- | --- |
| A1 | A5 | A9 | A13 |
| A2 | A6 | A10 | A14 |
| A3 | A7 | A11 | A15 |
| A4 | A8 | A12 | A16 |

编号为i的元素是明文中相应位置的字符。密钥也可以排成这种形式。

为获得良好的代数解释性，AES的所有运算都在GF(28)域上进行。上述矩阵中每个元素长度都是8位，恰好对应GF(28)域中的一个元素，可以在GF(28)域上进行运算。之后若无特殊说明，进行的运算都是这种运算。

记明文矩阵为A，密钥矩阵为K，密钥加法层的输出矩阵为A+K。当然，仅仅有这一步还是远远不够的，它甚至不能抵挡最简单的差分破解。因此我们还需要引入非线性。

#### 字节代换层

AES算法用S盒代换引入非线性。S盒将GF(28)域上的元素a进行一对一映射，具体操作为：

首先将a求逆得到a的逆元b。（求逆应用到扩展欧几里得算法，逻辑和RSA部分中的相同，这里不予赘述）

然后将b变为其向量表示，（小端序）在二元域上左乘变换矩阵S做线性变换。求逆本身已经是非线性变换，避免了差分破解，S又使得b的各位混在一起，增加安全性。为了避开不动点，在左乘变换矩阵后还需要加上一个偏置向量，得到最终的结果S(a)。

上述过程也可以通过事先生成S盒并查表完成，但这样容易受到缓存攻击，攻击者可以查看S盒各元素被查询的频率从而破译密钥。不过本次实验中为了仿真方便，我们仍然通过查表的方法实现了字节代换，S盒的生成代码在SBoxGeneration文件夹中。

#### 行位移层

经过密钥相加和字节代换，此时的密钥已经不易破解，但仍然局限于对单个符号的操作，容易通过概率分布破译。行位移层为列混淆层做准备，这两层主要引入雪崩特性，起到扩散效果。

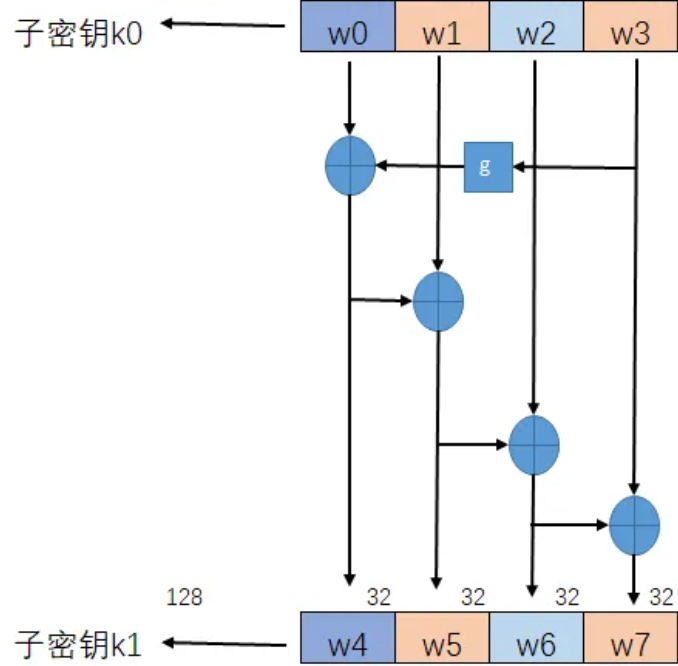
行位移层的操作比较简单。将经过之前两个步骤得到的矩阵A’第二行向左循环移位1列，第三行向左循环移位2列，第四行向左循环移位3列便完成了行位移操作。虽然简单，但行位移操作却非常重要，因为若没有行位移，不同行内的元素便无法通过列混淆混在一起。

#### 列混淆层

列混淆层将行位移得到的矩阵A’’左乘可逆混淆矩阵M得到输出。这一步将不同行、不同列的数据混在一起，完成了混淆操作。

#### 迭代

为了提升安全性，AES需要反复将上面几个步骤进行10次。为了消除AES加密算法中的对称性（上面用的混淆矩阵和S矩阵都是沿反对角线对称的），以及增强非线性，每轮密钥也要迭代生成，生成方式类似feistel加密，如下图[[3]](#footnote-3)：



以上的所有操作都是可逆的。AES解密只需反演AES加密的过程即可。本次实验的AES加密封装在AES.m中，AES解密封装在IAES.m中。

### 安全性与性能

性能上，AES加密的密钥量为2128，这使得其无法被暴力破解。（本次实验采用字符串密钥，但实际应用中应避免局限于字符串，因为会降低密钥量，且有意义的字符串更容易被猜到）同时，每次能加密的信息量为32byte，长度可以接受。此外，加密和解密需要的内存空间不大，速度也较快。

安全性上，AES非线性、非差分可破，雪崩效应显著，无法用统计方法破解。实际上，目前除旁路攻击外，对AES加密仍然没有足够可靠的破解方法。

## 四、DES加密算法

与AES相同，DES也是对称加密算法，每次对64bit明文进行加密。为实现非线性和雪崩效应，DES也采用了多轮置换/代换的结构，其中代换通过feistel结构完成。

### 算法及实现

首先对分组后64bit的明文进行IP置换。置换规则由映射S ({0,1,…63}->{0,1,…63})决定，置换后明文msg’满足：

之后将明文分为L0和R0两部分。

随后对密钥也进行类似的置换。DES加密中，密钥每一字节的最后一位为奇偶校验位，无需加以置换，只需置换剩下的56bit密钥位。对置换后的56位密钥，我们将其分成左右两个部分：C0和D0，每半边都有28位。

随后不断对C0和D0进行循环左移，每次左移1位直到左移16位，产生16对新的{Cn, Dn}。对每对{Cn, Dn}进行拼合，生成56位新密钥；随后对此新密钥进行变长置换，将56位缩减到48位，得到新密钥Kn。之后，DES算法要进行16次迭代，在第i次迭代中使用密钥Ki。

以Ln-1和Rn-1作为输入开始迭代，产生Ln和Rn。L0和R0由初始明文置换得来。

每一轮迭代中都进行这样的处理过程：

函数f的工作流程为：

* 将变长置换为48位序列E；
* 将E和Kn异或得到E’；
* 将E’分为8组E’n (n=1,2,…8)，每组6bit；
* DES算法中定义了8个映射S1~S8，均为{0,63}->{0,15}。将E’n (n=1,2,…8)替换为Sn(E’n ) (n=1,2,…8), 将结果缝合为E’’。
* 对E’’再次进行置换。将其分为8组，每组4bit，经置换后，各输入组的4个比特都被分配到不同的输出组之中，而且各输出组的4个比特也都来自不同的输入组。这样得到最终的结果。

以上的加密过程均为可逆的。只要反转密钥的使用顺序，自底向上重复加密流程即可解密。

### 安全性与性能

以现代的标准来看，DES密钥为56位，短于AES，加密复杂度也不及AES，容易被暴力计算破解。同时，DES每次加密的明文量也较小，使得扩散局限于较小的区域。但在AES出现前，设备算力水平较低，DES一度是主流的数据加密方式。

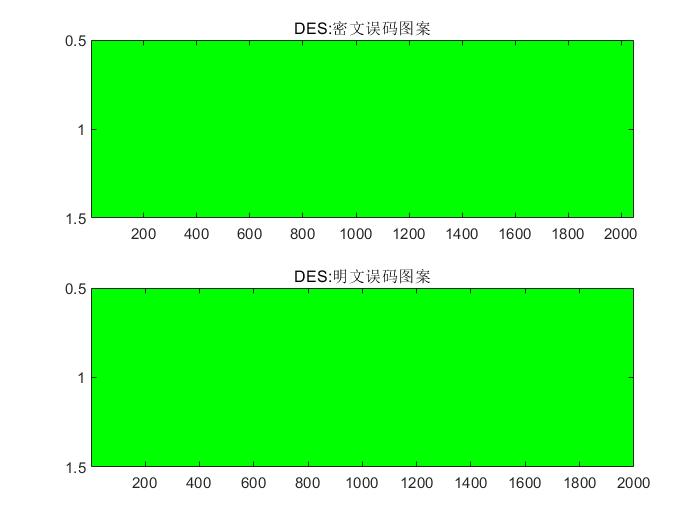
## 五、实验验证

首先验证加密架构的正确性。

#### RSA+AES，无传输错误：

图表, 条形图

描述已自动生成



在密钥和密文都正确的情况下，解密不会出现错误。

其次验证安全性：

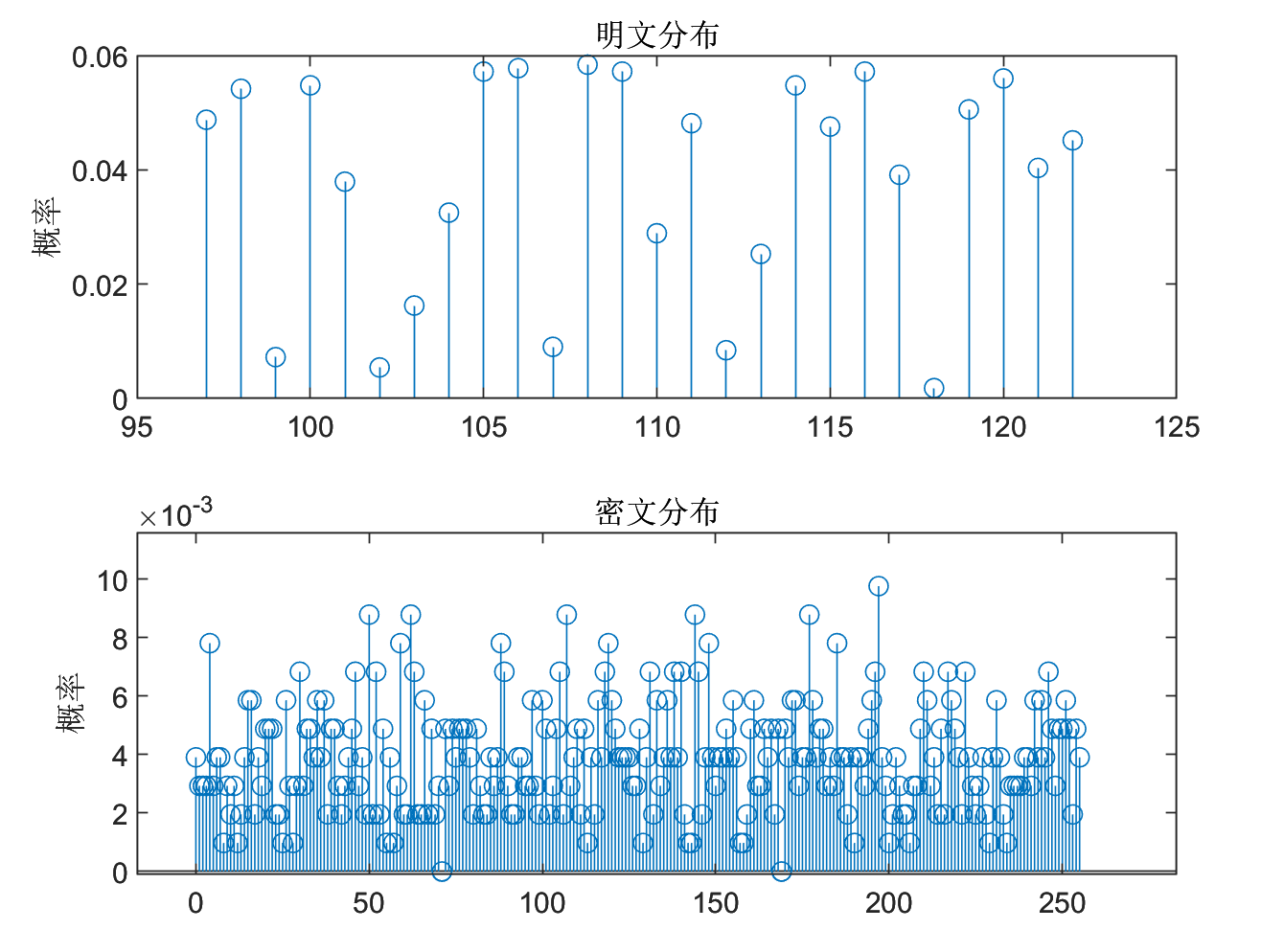
#### 仅AES密文错误：

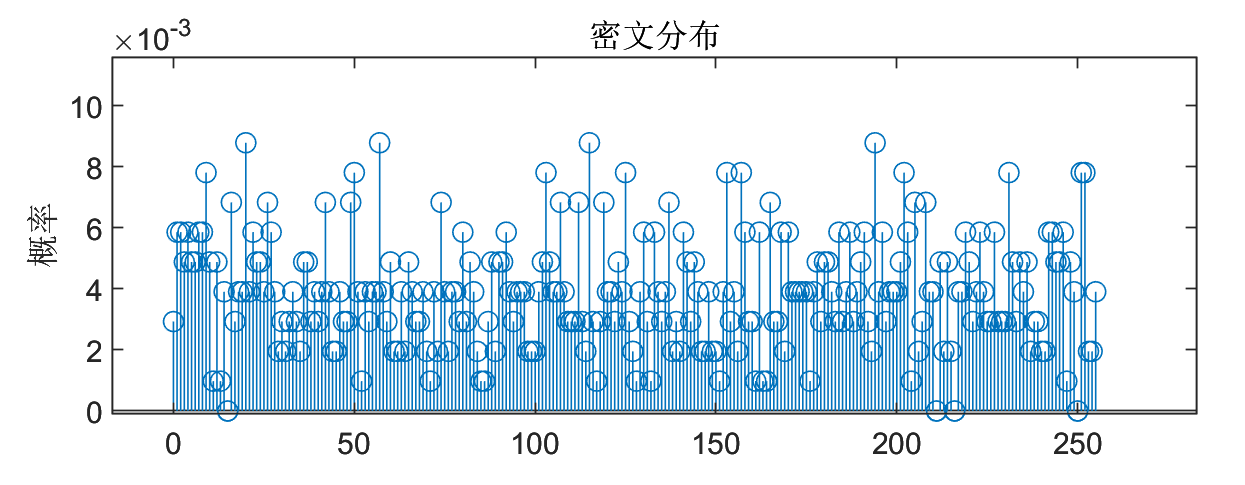
图片包含 图表

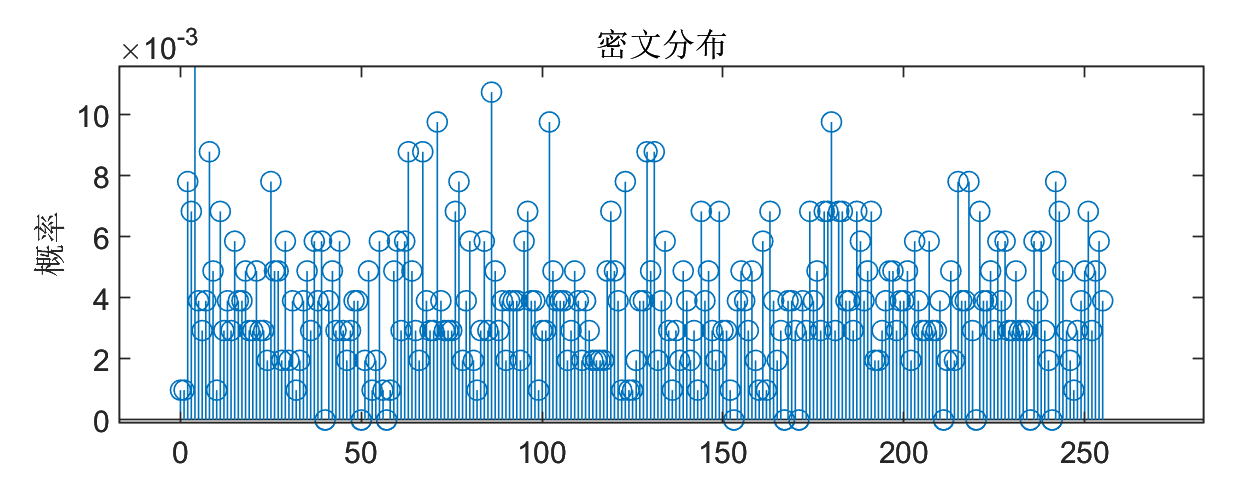
描述已自动生成

AES的雪崩效应使得密文中少量的错误会扩散到整个加密块中，这有助于应对选择明文攻击。另外，上图也反映出雪崩效应受能加密的明文长度的限制，在性能一定的前提下，较好的加密算法应当能够一次性加密足够长的明文。

为验证AES的扩散能力，我们对AES明文和密文字符分布进行了统计，多次实验的结果如下：







横坐标为字符ASCII码，纵轴为出现频率。在明文只出现在指定区域(26个英文字母)且方差较大的情况下，密文方差较小，在大部分位置几乎均一分布。多次测试中，密文分布不断变化，看不出明显的规律，说明扩散效应相当好，足以抵御统计攻击。

#### 仅RSA密文错误：

图表, 条形图

描述已自动生成

由于RSA将所有128位明文作为整体进行模运算，其加密也具有雪崩效应。上图中，RSA密文仅有5bit的错误，而明文有约50%出错。不过，由于非对称加密分离了公钥和私钥，加密密钥被公开，因此这种雪崩效应对安全性贡献不大，其更大的启示是，在传输RSA密文时应当通过校验位、ARP等机制提高正确传输的概率。

#### 仅DES密文错误：

图表, 条形图

描述已自动生成

与AES类似，DES也具有良好的雪崩效应。

# 三、附录

## 成员分工：

邵晨扬：信道绘图，DES加解密，安全、信道编码联调。

唐钰凯：

张鹤龄：RSA和AES编写/调试

## 文件清单：[[4]](#footnote-4)

1. n非常大时，函数返回的Pu和Pr可能不满足PuPr mod Φ(n) = 1，而满足PuPr mod Φ(n) = -1，推测可能由于舍入误差导致。因此，在产生Pu和Pr后，还需进一步筛选得到满足PuPr mod Φ(n) = 1的Pu和Pr。 [↑](#footnote-ref-1)
2. RSA的安全关键在于大素数的产生。只要产生的随机数在2^511~2^512-1均匀分布，就能保证最大的安全性；反之如果随机算法（或伪随机算法）无法保证均匀分布，密钥就有可能被根据随机数的概率分布破解。 [↑](#footnote-ref-2)
3. [密码学基础：AES加密算法 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/78913397) [↑](#footnote-ref-3)
4. 本次实验的所有代码可在[IntroductionOfCoding/Task2 at main · AllergoP/IntroductionOfCoding (github.com)](https://github.com/AllergoP/IntroductionOfCoding/tree/main/Task2)获取。 [↑](#footnote-ref-4)