



北京理工大学
Beijing Institute of Technology

本科生实验报告

课程名称: 高级数字信号处理
实验名称: 信号检测实验

任课教师:	田黎育		实验教师:	田黎育			
实验日期:	2025 年 11 月 10 日		实验地点:				
实验类型:	<input checked="" type="checkbox"/> 原理验证		<input type="checkbox"/> 综合设计	<input type="checkbox"/> 自主创新			
学生姓名:	闫子易	班级:	13212302	学号:	1120230621		
学院:	集成电路与电子学院		专业:	电子科学与技术			
组号:		同组同学:					
成绩:							



集成电路与电子学院
SCHOOL OF INTEGRATED CIRCUITS
AND ELECTRONICS

一、实验目的

复高斯白噪声（均值为 0，方差为 1）求模后 10^{-6} 虚警率，门限怎么设置？

- ① 10^6 个数，求模后分布图,然后找极大值,及其位置。门限在最大值与次大值之间
- ② 10 次 10^6 个数 10 个最值,观察规律
- ③ 100 次 10^6 个数 100 个最值，观察规律
- ④ CFAR 200 点 复信号 $A \cdot e^{j\theta} +$ 复噪声 检测概率 $\geq 85\%$, 且虚警率小于 10^{-6} ,目标幅度是多少。做 $10^6/200=5000$ 次。

CFAR 左右各取 10 个训练单元（共 20 个）

rgn(2)固定种子

二、实验原理

本实验主要研究复高斯白噪声在取模后的统计特性，并结合恒虚警检测(CFAR)方法来分析门限设置与虚警率、检测概率之间的关系。复高斯白噪声的实部和虚部均服从均值为零、方差为一的高斯分布，取模后幅度服从 Rayleigh 分布。通过大量样本（如 10^6 个）可以得到其幅度分布的直方图，并与理论瑞利分布进行对比，从而验证噪声的统计规律。

在虚警率分析中，利用极值理论：在有限样本中，噪声的最大值和次大值之间的差距可以用来估计门限。如果门限设置在最大值与次大值之间，就能控制虚警率在 10^{-6} 量级。通过重复试验（如 10 次、100 次），观察最大值、次大值和虚警率的统计规律，可以验证门限的稳定性和虚警率的分布情况。

在目标检测部分，采用 CA-CFAR 算法。具体做法是：将数据分组，每组长度固定（200 点），在中心单元加入一个复目标信号 $Ae^{j\theta}$ ，其余为复高斯噪声。通过左右各取若干训练单元（如 20 个），估计噪声功率，并利用 CFAR 理论公式计算阈值常数 α ，得到检测门限。CUT 单元的功率若超过门限，则判定为检测到目标。通过蒙特卡洛实验统计检测概率 P_d ，并与设定的虚警率 $P_{fa} = 10^{-6}$ 对比，进而搜索满足 $P_d \geq 0.85$ 的最小目标幅度。

三、运行代码及实验内容

1. 复高斯白噪声 10^6 个数，求模后分布图

```
% Distribution_of_the_Magnitude_of_Complex_Gaussian_White_Noise

clc; clear; close all;

% 固定随机数种子
rng(2);

% 生成  $10^6$  个复高斯白噪声
N = 1e6;
x = randn(N,1) + 1i*randn(N,1);

% 取模
r = abs(x);

% 绘制分布图（直方图）
figure;
h = histogram(r, 'Normalization', 'pdf', 'BinWidth', 0.05);
xlabel('幅度');
ylabel('概率密度');
title('复高斯白噪声取模后的分布');
hold on;

% 理论分布对比: Rayleigh 分布
sigma = 1; % randn 默认方差=1
r_vals = linspace(0,6,200);
pdf_rayleigh = (r_vals./sigma.^2).*exp(-r_vals.^2/(2*sigma.^2));
plot(r_vals, pdf_rayleigh, 'r', 'LineWidth', 2);
legend('模拟结果','Rayleigh 理论分布');

% 计算直方图各 bin 的中心和最大值位置
edges = h.BinEdges;
centers = (edges(1:end-1) + edges(2:end)) / 2;
[peakVal, idxPeak] = max(h.Values);
peakX = centers(idxPeak);
peakY = peakVal;

% 在图中标注最高点坐标
plot(peakX, peakY, 'ko', 'MarkerFaceColor', 'y', 'MarkerSize', 8);
```

```

labelStr = sprintf('(%0.3f, %0.3f)', peakX, peakY);
% 将文字放在点的右上方, 避免与曲线重叠
textOffsetX = 0.15; % 横向偏移
textOffsetY = 0.05 * max(h.Values); % 纵向偏移
text(peakX + textOffsetX, peakY + textOffsetY, labelStr, 'FontSize', 10,
'FontWeight', 'bold');

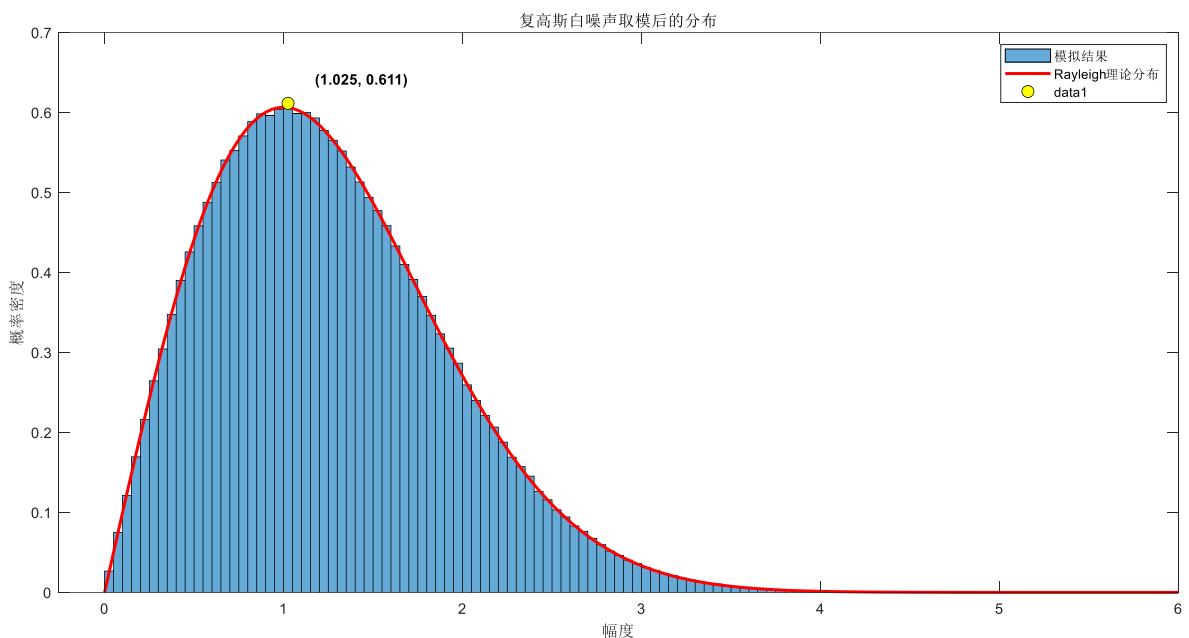
hold off;

```

首先, 通过 `rng(2)` 固定随机数种子, 这样每次运行都会得到相同的结果, 保证可重复性。接着用 `randn` 生成 10^6 个复高斯白噪声样本, 实部和虚部都是独立的标准正态分布。然后取模得到幅度, 这一步就是把复数的大小提取出来。

在绘图部分, 先用直方图展示这些幅度的分布, 并且归一化成概率密度的形式。这里设置了 `BinWidth=0.05`, 让分布看起来比较平滑。接着, 代码叠加了理论的 Rayleigh 分布曲线, 把它们画在同一个图里做对比。

之后, 找到直方图的最高点, 也就是概率密度的峰值位置。通过 `h.BinEdges` 算出每个 bin 的中心, 再用 `max(h.Values)` 找到最大值对应的索引。



分布图展示了大量复高斯白噪声取模后的模拟结果。

从零开始, 概率密度在 $r=0$ 时为零。

随着 r 增大, 概率密度迅速上升, 在某个正值附近达到峰值。

之后逐渐下降，尾部呈现**指数衰减**。

形状类似一个**单峰的右偏曲线**：左边陡峭上升，右边缓慢下降。

我在直方图上叠加了一条红色的平滑曲线，这是理论上的 Rayleigh 分布。它和直方图的形状非常吻合，说明模拟结果与理论模型一致。图中还用一个黄色圆点标出了直方图的最高点，并在旁边标注了坐标值(1.025,0.611)。

直方图和曲线直观地验证了复高斯白噪声取模后确实服从 Rayleigh 分布。

但是为什么会和 Rayleigh 分布的曲线重叠呢？高斯白噪声和瑞利分布又有什么紧密的关系？

这就得从数学原理入手。

复高斯白噪声通常表示为：

$$x = x_r + ix_i$$

其中： $x_r \sim \mathcal{N}(0, \sigma^2)$ - $x_i \sim \mathcal{N}(0, \sigma^2)$

并且 x_r, x_i 相互独立。

我们关心的是它的模：

$$r = |x| = \sqrt{x_r^2 + x_i^2}$$

1. (x_r, x_i) 的联合概率密度函数为：

$$f_{x_r, x_i}(u, v) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{u^2 + v^2}{2\sigma^2}\right)$$

这是一个二维零均值、方差为 σ^2 的高斯分布。

2.转换到极坐标

令：

$$u = r\cos\theta, \quad v = r\sin\theta$$

则雅可比行列式为：

$$J = r$$

于是联合分布在极坐标下为：

$$f_{r,\theta}(r, \theta) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) \cdot r$$

3. 边缘分布（模的分布）

对角度 θ 积分：

$$f_r(r) = \int_0^{2\pi} f_{r,\theta}(r, \theta) d\theta = \int_0^{2\pi} \frac{1}{2\pi\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) \cdot r d\theta$$

得到：

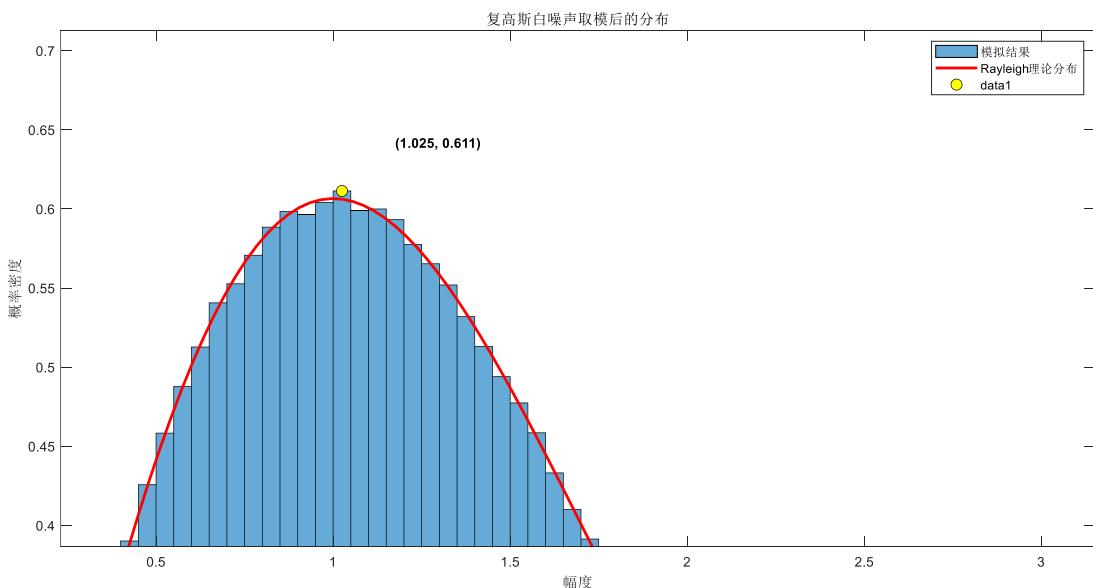
$$f_r(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right), \quad r \geq 0$$

这个分布正是 **Rayleigh 分布**，其概率密度函数为：

$$f(r; \sigma) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right), \quad r \geq 0$$

因此，复高斯白噪声的模服从 Rayleigh 分布。

在得知复高斯白噪声的模与 Rayleigh 分布的关系后，我又在代码中把最大值的点标注了出来，发现其坐标为 (1.025, 0.611)。我又产生了一个疑问，最高点的坐标为什么是这个呢？



复高斯白噪声取模后服从 Rayleigh 分布：

$$f(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right), \quad r \geq 0$$

要找到它的最大值（峰值），我们对 $f(r)$ 求导并令其为零：

$$\frac{d}{dr} f(r) = \frac{1}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) \left(1 - \frac{r^2}{\sigma^2}\right) = 0$$

解得：

$$r_{\text{peak}} = \sigma$$

当 $\sigma = 1$ 时，峰值位置应在：

$$r_{\text{peak}} = 1$$

对应的概率密度值为：

$$f(r_{\text{peak}}) = \frac{1}{\sigma} \exp(-1/2) = e^{-0.5} \approx 0.607$$

直方图标注的最大值坐标是：

$$(1.025, 0.611)$$

这与理论值：

$$(1.000, 0.607)$$

非常接近。

为什么会有微小偏差？

因为生成了 10^6 个样本，虽然很多，但仍然是有限样本量，存在统计波动。并且直方图的 bin 宽度设为 0.05，峰值位置只能落在某个 bin 的中心，而不是连续的精确值。

2.

10⁶ 个数，求模后分布图,然后找极大值,及其位置。门限在最大值与次大值之间

10 次 10⁶ 个数 10 个最值，观察规律

100 次 10⁶ 个数 100 个最值，观察规律

观察到这个实验其实 1 次 10⁶、10 次 10⁶、100 次 10⁶ 主体操作都是差不多的，代码能有很大的复用部分。所以为了代码的简洁，我在开头单独设了一个地方存放实验次数，这样就不用 1, 10, 100 写三个脚本了。

% 可修改的参数

```
N = 1e6; % 每次试验的样本点数  
numTrials = 100; % 设置为 1、10、100 或其他正整数（控制实验次数）
```

% 预分配

```
maxVals = zeros(1, numTrials);  
secondMaxVals = zeros(1, numTrials);  
thresholds = zeros(1, numTrials);  
falseAlarms = zeros(1, numTrials);  
falseAlarmRates = zeros(1, numTrials);
```

每次试验先生成 N 个复高斯噪声样本并取其幅值，然后提取最大值和次大值，以两者的中点作为门限。接着统计超过门限的样本点数，计算虚警率，并将结果记录下来。经过多次试验后，程序对最大值、次大值、门限和虚警率进行均值与标准差的汇总，并绘制曲线以直观展示规律。

并且设置，如果实验次数仅为 1 次，就绘制该次试验最大值附近的样本。大于 1 次的就绘制每次试验的最大值/次大值/门限，找规律。同时，也在命令行打印每次实验的最大值、索引、虚警率（当然由于门限设为最大值与次大值之间，所以每次实验虚警率恒为 10⁻⁶），还有最大值、次大值、门限的所有实验的均值、标准差。

① 1 次 10⁶

```
Trial 1/1: 最大值 = 5.184576 (索引 399290), 次大值 = 5.084631, 门限 = 5.134603, 虚警数 = 1, 虚警率 = 1.000000e-06
```

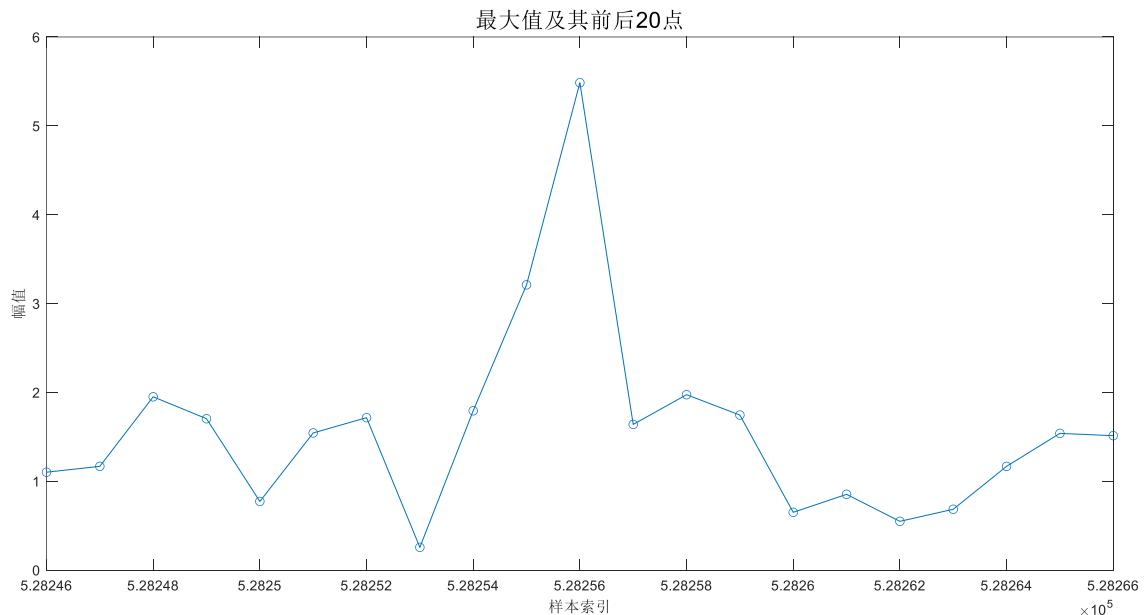
==== 汇总 (1 次试验, 每次 N = 1000000) ====

最大值均值 = 5.184576, 标准差 = 0.000000

次大值均值 = 5.084631, 标准差 = 0.000000

门限均值 = 5.134603, 标准差 = 0.000000

虚警率均值 = 1.000000e-06, 标准差 = 0.000000e+00



可以看到, 最大值和次大值的差距在 0.1 左右, 符合随机噪声的统计特性。最大值为 5.184576, 对应索引位置为 399290。

因为总样本数是 10^6 , 只有 1 个点超过门限, 所以虚警率正好是 $1/N$ 。这与理论预期完全一致。

最大值及其前后 20 点可以看到峰值点是主要的异常或误差来源, 其他点虽然也有误差, 但影响相对有限。

② 10 次 10^6

Trial 1/10: 最大值 = 5.184576 (索引 399290), 次大值 = 5.084631, 门限 = 5.134603, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 2/10: 最大值 = 5.484916 (索引 528256), 次大值 = 5.401522, 门限 = 5.443219, 虚警数 = 1, 虚警率 =

1.000000e-06

Trial 3/10: 最大值 = 5.239038 (索引 684607), 次大值 = 5.029693, 门限 = 5.134365, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 4/10: 最大值 = 5.291629 (索引 851376), 次大值 = 5.127636, 门限 = 5.209633, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 5/10: 最大值 = 5.454909 (索引 678579), 次大值 = 5.132025, 门限 = 5.293467, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 6/10: 最大值 = 5.565119 (索引 611944), 次大值 = 5.042453, 门限 = 5.303786, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 7/10: 最大值 = 5.648761 (索引 341556), 次大值 = 5.181283, 门限 = 5.415022, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 8/10: 最大值 = 5.522296 (索引 987588), 次大值 = 5.215067, 门限 = 5.368682, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 9/10: 最大值 = 5.728452 (索引 861283), 次大值 = 5.103842, 门限 = 5.416147, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 10/10: 最大值 = 5.118432 (索引 618020), 次大值 = 5.067271, 门限 = 5.092851, 虚警数 = 1, 虚警率 = 1.000000e-06

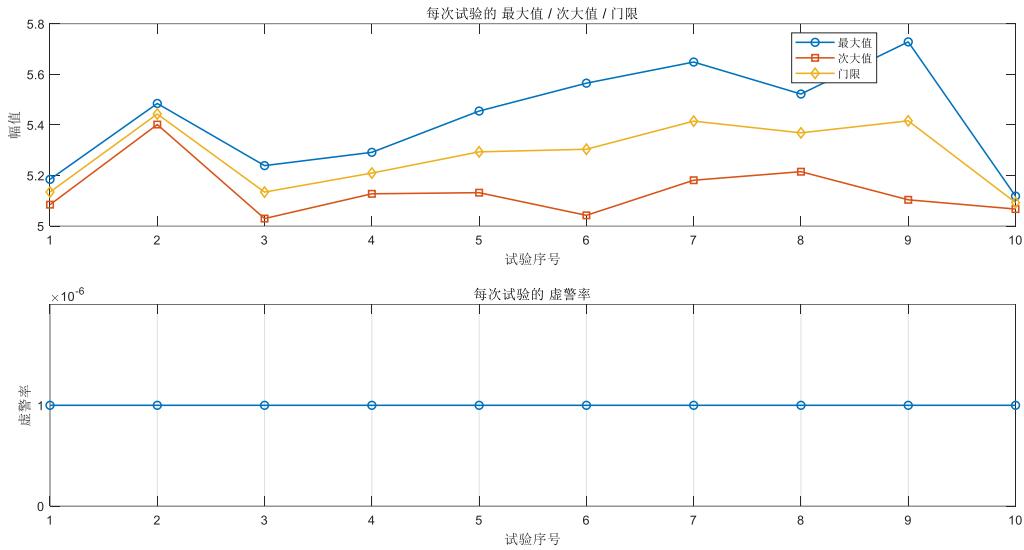
==== 汇总 (10 次试验, 每次 N = 1000000) ====

最大值均值 = 5.423813, 标准差 = 0.205448

次大值均值 = 5.138542, 标准差 = 0.109176

门限均值 = 5.281177, 标准差 = 0.130950

虚警率均值 = 1.000000e-06, 标准差 = 2.232128e-22



每次试验的最大值大约在 5.1~5.7 之间，均值约 5.42，标准差约 0.20。这说明在百万点复高斯噪声幅值中，极大值的波动范围有限，集中在一个稳定区间。

次大值分布：次大值基本在 5.0~5.4 之间，均值约 5.14，标准差约 0.11。相比最大值，它更集中，说明极值之外的次极值差异较小。

门限是最大值与次大值的中点，均值约 5.28，标准差约 0.13。门限始终落在最大值和次大值之间，且波动幅度比最大值小。

每次试验虚警数都为 1，虚警率固定在 10^{-6} 。这是因为门限设置在最大值和次大值之间，只有最大值超过门限，因此虚警数恒为 1。标准差几乎为零，说明虚警率完全由算法逻辑决定，而不是随机波动。

③ 100 次 10^6

```
Trial 93/100: 最大值 = 5.721347 (索引 322274), 次大值 = 5.513623, 门限 = 5.617485, 虚警数 = 1, 虚警率 = 1.000000e-06
```

```
Trial 94/100: 最大值 = 5.424143 (索引 842569), 次大值 = 5.269816, 门限 = 5.346979, 虚警数 = 1, 虚警率 = 1.000000e-06
```

```
Trial 95/100: 最大值 = 5.052034 (索引 859553), 次大值 = 4.961080, 门限 = 5.006557, 虚警数 = 1, 虚警率 = 1.000000e-06
```

```
Trial 96/100: 最大值 = 5.203685 (索引 630859), 次大值 = 4.973644, 门限 = 5.088664, 虚警数 = 1, 虚警率 =
```

1.000000e-06

Trial 97/100: 最大值 = 5.092100 (索引 486493), 次大值 = 4.942982, 门限 = 5.017541, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 98/100: 最大值 = 5.159072 (索引 506456), 次大值 = 5.141347, 门限 = 5.150210, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 99/100: 最大值 = 5.299969 (索引 301158), 次大值 = 5.258336, 门限 = 5.279153, 虚警数 = 1, 虚警率 = 1.000000e-06

Trial 100/100: 最大值 = 5.249287 (索引 171280), 次大值 = 5.145227, 门限 = 5.197257, 虚警数 = 1, 虚警率 = 1.000000e-06

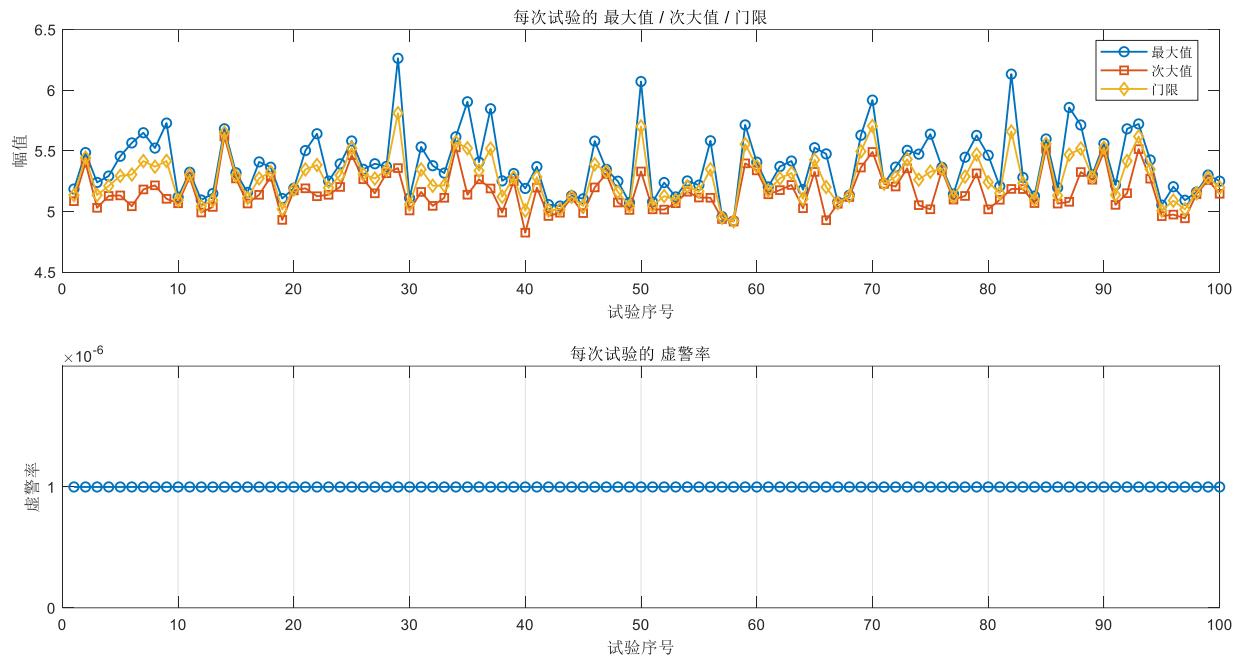
==== 汇总 (100 次试验, 每次 N = 1000000) ===

最大值均值 = 5.383842, 标准差 = 0.259652

次大值均值 = 5.164239, 标准差 = 0.156240

门限均值 = 5.274040, 标准差 = 0.186292

虚警率均值 = 1.000000e-06, 标准差 = 4.256501e-22



这次把实验次数增加到 100, 可以看到一些新的特点。最大值的标准差从之前的约 0.20 增加到 0.26, 次大值的标准差也从 0.11 增加到 0.16, 门限的标准差从 0.13 增

加到 0.19。说明随着试验次数增多，极值的波动范围被更完整地采样出来，分布的离散性更明显。虚警率仍然严格保持在 10^{-6} ，标准差依旧接近零。

增加实验次数后，结果更能反映极值统计的真实分布，最大值和次大值的波动范围更宽，均值有轻微调整，但整体规律保持一致。换句话说，试验次数越多，统计结果越接近理论分布，差异主要体现在标准差的扩大和均值的微调。

3. CFAR 200 点 复信号 $A \cdot e^{j\theta}$ +复噪声 检测概率>=85%，且虚警率小于 10^{-6} ，目标幅度是多少。做 $10^6/200=5000$ 次。

代码先统一设置总样本数与分组长度、训练单元数与虚警率。

```
N = 1e6; % 总样本数
groupLen = 200; % 每组长度
numGroups = N / groupLen; % 应为 5000
if mod(numGroups,1) ~= 0
    error('N must be divisible by groupLength.');
end

% CFAR 训练单元设置
numTrainLeft = 10; % 左侧训练单元数
numTrainRight = 10; % 右侧训练单元数
numTrain = numTrainLeft + numTrainRight; % 总训练单元数 (应为 20)

% CFAR 虚警率 (每个 CUT 的目标误警概率)
Pfa_desired = 1e-6; % 可调整 (默认 1e-6)

% 目标幅度设置
A_fixed = 5; % 如果不自动搜索，直接使用此 A
```

并采用了 CA-CFAR 算法，左右各取 10 个训练单元，总共 20 个；阈值因子 alpha 由理论公式计算，保证在给定虚警率 Pfa 下检测门限合理。这样做的好处是噪声功率可能随时间或空间变化，直接用周围单元估计噪声能自适应调整门限，从而保持恒定的虚警率。

```
numTrain = numTrainLeft + numTrainRight;
```

```

alpha      = numTrain * (Pfa_desired^(-1/numTrain) - 1);
centerIdx = ceil(groupLen/2);

% 噪声 + 目标
w = (randn(1, groupLen) + 1i*randn(1, groupLen)) / sqrt(2);
theta = 2*pi*rand();
w(centerIdx) = w(centerIdx) + A*exp(1i*theta);

% 训练单元素引
idxLeft  = (centerIdx - numTrainLeft):(centerIdx - 1);
idxRight = (centerIdx + 1):(centerIdx + numTrainRight);
idxLeft  = idxLeft(idxLeft >= 1 & idxLeft <= groupLen);
idxRight = idxRight(idxRight >= 1 & idxRight <= groupLen);
trainingIdx = [idxLeft, idxRight];

% 噪声估计与判决
P_avg     = mean(abs(w(trainingIdx)).^2);
threshold = alpha * P_avg;
CUT_power = abs(w(centerIdx))^2;
isDetect  = (CUT_power > threshold);

```

`run_cfar_once(A,...)`: 对每组生成复高斯噪声, CUT 位置叠加幅度为 A 的随机相位复目标, 选取左右训练单元计算平均功率 P_avg, 阈值为 alpha*P_avg, 比较 CUT 功率得到检出并统计 Pd; 函数可选保存一个随机组的样本与判决数据作为示例, 确保绘图来自真实统计试验。固定 A 进行一次完整蒙特卡洛以得到实际 Pd 并同步保存示例。最后随机挑选 5000 次实验中的一次, 绘图, 分别展示幅值域和功率域, 标出待检单元 CUT, 就是每组样本中要进行检测的那个点——这里放在索引 100 的位置, 并用红色竖线标出, 训练单元与阈值, 便于直观验证检测逻辑。额外细节包括中心 CUT 索引、边界训练单元保护、随机种子控制重现性、进度与日志输出, 以及以功率域检测并在幅值图中用 `sqrt(threshold)` 对应阈值线可视化。

- 但我又发现, 其实给定 $P_d \geq 85\%$, 那么 A 其实是能算出一个大概的范围的。

检测单元含目标: $x = Ae^{j\theta} + n$, 其中 n 为复高斯白噪声。检测统计的幅度为 Rician 分布。若噪声每维方差为 σ^2 , 则等效噪声功率为 $2\sigma^2$ 。常用线性信噪比

$$\text{SNR} = \frac{A^2}{2\sigma^2}.$$

在 $P_{fa} = 10^{-6}$ 的虚警率约束下，经典雷达检测结果表明要实现 $P_d \approx 0.85 \sim 0.90$ 往往需要约 12–13dB 的 SNR；文献在 $P_{fa} = 10^{-6}, P_d = 90\%$ 条件下给出所需 $\text{SNR} \approx 12.8\text{dB}$ 的量级。代入上式可得到目标幅度的理论量级：

$$\text{SNR}_{\text{lin}} = 10^{12.8/10} \approx 19.05, \quad A = \sigma \sqrt{2 \text{SNR}_{\text{lin}}} \approx \sigma \cdot 6.17.$$

所以在 CFAR 设置与 $P_{fa} = 10^{-6}$ 的约束下，理论上 A 的量级约为 6.2σ 左右可达到约 85–90% 的检测概率。若严格锁定 $P_d \geq 85\%$ ，预计 $A \approx (5.7 \sim 6.1)\sigma$ 。

也就是说，我们不用自己一个一个手动修改 A 值观察 P_d 是否达到了 85%，这样不仅麻烦，重复无意义操作，还不精准。所以我直接给程序一个大概的 A 的范围，然后用二分法查找刚好使 $P_d > 85\%$ 的那个 A 值，这就是实验最后要的 A。

```

iter = 0;
while (Amax - Amin) > tolA && iter < max_search_iter
    A_mid = (Amin + Amax) / 2;
    Pd_mid = run_cfar_once(A_mid, numGroups, groupLen, numTrainLeft,
                           numTrainRight, alpha, showProgress, false);
    if verbose
        fprintf('iter %d: A_mid = %.6f, Pd = %.4f\n', iter, A_mid, Pd_mid);
    end
    if Pd_mid >= Pd_target
        Amax = A_mid; % 可接受，尝试更小的 A
    else
        Amin = A_mid; % 不足，增大 A
    end
    iter = iter + 1;
end
A_found = Amax;
% 最后一次用于报告 Pd 并保存示例 (saveExample = true)
[Pd_found, example] = run_cfar_once(A_found, numGroups, groupLen,
                                      numTrainLeft, numTrainRight, alpha, showProgress, true);
fprintf('\n搜索完成：找到近似最小 A = %.6f 使 Pd >= %.2f, 实际 Pd = %.4f\n',
       A_found, Pd_target, Pd_found);
else
    A_found = A_fixed;

```

```

[Pd_found, example] = run_cfar_once(A_found, numGroups, groupLen,
numTrainLeft, numTrainRight, alpha, showProgress, true);
fprintf('使用固定 A = %.6f, 测得 Pd = %.4f\n', A_found, Pd_found);

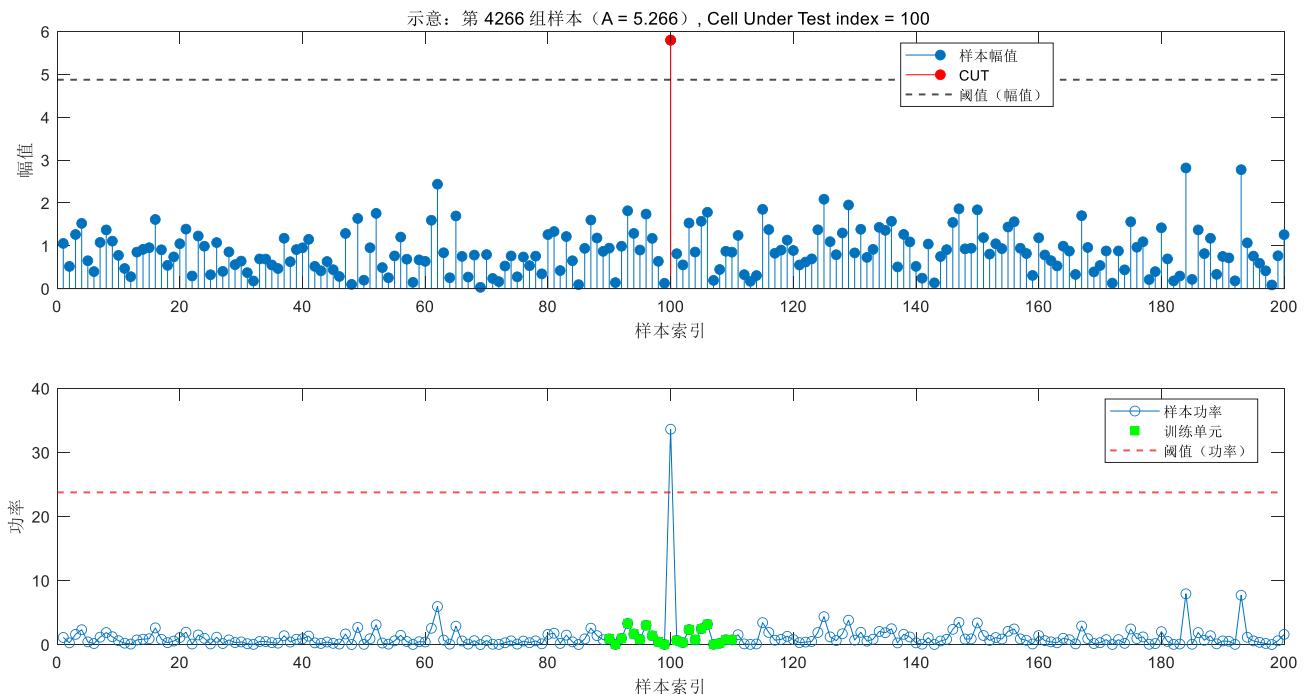
```

使用了自动查找后，脚本在 2s 内就找到了比较精准的幅度 A。

processed 4500/5000 groups

processed 5000/5000 groups

搜索完成：找到近似最小 $A = 5.266113$ 使 $Pd \geq 0.85$ ，实际 $Pd = 0.8590$



图中展示了其中一次典型的 CFAR 检测结果。上半部分以幅度为纵轴，所有样本的幅值以蓝色圆点标出，中心位置的第 100 个样本用红色竖线强调，幅度明显高于周围噪声。黑色虚线代表由训练单元平均功率计算出的阈值，位置低于该目标幅值，说明在这一组数据中目标被成功判出。下半部分以功率为纵轴，蓝色曲线描绘了各样本的功率分布，绿色标记突出显示了用于估计背景的训练单元，红色虚线表示功率阈值。第 100 个样本（也就是目标信号）的功率峰值远高于阈值，形成清晰的峰值。

那么我们把 A 改小， <5.266113 ，比如设为 4，关闭自动计算功能，`autoFindA = false`。

```
processed 3500/5000 groups  
processed 4000/5000 groups  
processed 4500/5000 groups  
processed 5000/5000 groups|
```

使用固定 $A = 4.000000$, 测得 $P_d = 0.3338$

`x >>`

可以看到，虽然 $A=4$ 已经放大很多了，并且比 5.266 就小了一点，但 P_d 检测概率只有 0.3338，远小于要求的 85%。

再把 A 改大， >5.266113 ，比如设为 6，关闭自动计算功能，`autoFindA = false`。

```
processed 3500/5000 groups  
processed 4000/5000 groups  
processed 4500/5000 groups  
processed 5000/5000 groups
```

使用固定 $A = 6.000000$, 测得 $P_d = 0.9656$

`x >>`

可以看到这时 P_d 检测概率已经达到了 0.9656，远超 85%。

```
processed 3500/5000 groups  
processed 4000/5000 groups  
processed 4500/5000 groups  
processed 5000/5000 groups
```

使用固定 $A = 8.000000$, 测得 $P_d = 1.0000$

`>>`

若将 A 改为 8 及以上， P_d 检测概率就已经达到了 100%，也就是完全能检测正确了。

实验收获和体会

这次实验让我对复高斯白噪声和 Rayleigh 分布的关系有了更直观的认识。以前在书本上看到公式时，总觉得只是推导上的结果，但当我真正用代码生成百万级的噪声样本并画出直方图，再和理论曲线对比时，那种几乎重合的感觉让我很震撼。尤其是峰值位置和理论值的差距只有很小的一点，我第一次体会到“理论和实验吻合”不是一句空话，而是可以在数据里清晰看到的。通过多次实验，我也发现极值的波动范围并不是随意的，而是有一定规律，最大值和次大值之间的差距大概在 0.1 左右，这让我理解了为什么门限可以设在两者之间来保证虚警率在 10^{-6} 的量级。

CFAR 检测的目标幅度搜索是我觉得最有挑战性也最有收获的部分。一开始我还想着手动修改 A 值来试检测概率，后来发现既麻烦又不精准，改用二分法后，短短 2 秒就找到了满足要求的最小 A 值（5.266 左右），这让我明白解决问题要找对方法，不能蛮干。更关键的是，先通过理论估算出 SNR 范围（12–13dB），再确定 A 的大致量级（6.2 σ 左右），再用蒙特卡洛实验验证，这种“先理论预判再实践验证”的思路特别实用，让后续的搜索有了明确方向，不会盲目尝试。另外，理解了 CFAR 算法中“用周围训练单元估计噪声功率”的逻辑后，我才明白为什么自适应门限比固定门限更稳定——毕竟实际场景中噪声不会一成不变，这种贴合实际的设计思路也让我对“恒虚警检测”的意义有了更深的理解。

整个实验过程中，我最大的收获是把理论和实践结合起来的体验。推导公式时觉得抽象，但在代码和数据里看到它们的体现，就会觉得这些知识是有用的、能落地的。同时也让我意识到，实验设计里很多细节，比如随机种子的固定、训练单元的选择、样本数量的大小，都会直接影响结果的稳定性和可信度。做完实验，我对信号检测的统计规律和 CFAR 算法的意义有了更深的理解，也更清楚为什么在工程应用里必须考虑虚警率和检测概率的平衡。