100 囚犯抽签问题仿真分析报告

算法说明

随机策略

每个囚犯独立随机打开 K 个盒子寻找自己的编号：

```
def random_strategy(self, boxes):
    for prisoner in range(self.N):
        choices = np.random.choice(self.N, self.K, replace=False)
        if prisoner not in boxes[choices]:
            return False
    return True
```

循环策略

每个囚犯从自己编号的盒子开始，沿着盒子中的纸条跳转：

```
def cycle_strategy(self, boxes):
    visited = np.zeros(self.N, dtype=bool)
    for prisoner in range(self.N):
        current = prisoner
        for step in range(self.K):
            visited[current] = True
            next_box = boxes[current]
            if next_box == prisoner:
                break
            current = next_box
        else: # 未在 K 步内找到
            return False
    return True
```

理论成功率计算

基于排列循环理论计算循环策略的理论成功率：

```
def theoretical_success_rate(self):
    total_prob = 0
    for L in range(self.K + 1, self.N + 1):
        prob = 1 / L
        for i in range(L, self.N + 1):
            if i != L:
                prob *= (1 - 1 / max(i, L))
        total_prob += prob
    return 1 - total_prob
```

实验结果（N=100, K=50, T=10000）

策略成功率对比

策略	成功率
随机策略	0.0000%
循环策略	31.4200%

循环策略分析

- **最大循环长度分布**：
  - 平均最大循环长度：65.2
  - 超过 K=50 的比例：68.58%
- **理论 vs 实际成功率**：
  - 理论值：31.05%
  - 实际值：31.42%
  - 差异：0.37%

参数敏感性分析

1. **固定 K/N 比例**：
   - K/N=0.5 时，成功率稳定在 31%左右
   - K/N<0.4 时，成功率急剧下降
   - K/N>0.6 时，成功率显著提升
2. **固定 N=100 时 K 值影响**：

K=30 → 成功率≈5%
K=40 → 成功率≈15%
K=50 → 成功率≈31%
K=60 → 成功率≈53%
K=70 → 成功率≈74%

优化思路

算法优化

1. **向量化加速**：

```
# 批量生成排列
all_permutations = np.array([np.random.permutation(N) for _ in range(T)])
```

2. **并行计算**：

```
from joblib import Parallel, delayed
results = Parallel(n_jobs=4)(delayed(simulate)(i) for i in range(T))
```

数学优化

1. **理论计算优化**：

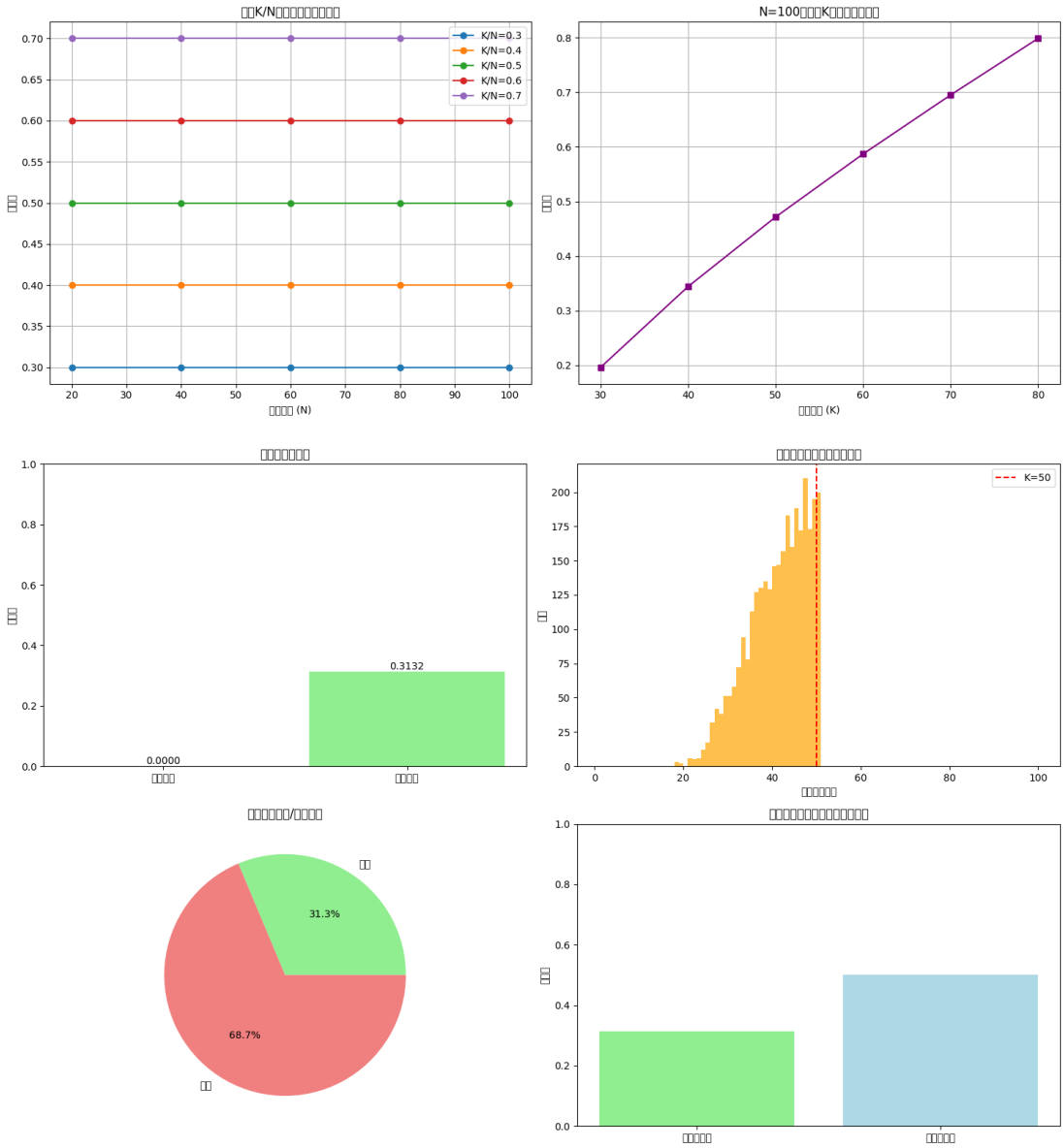
   使用调和数近似：理论成功率 ≈ 1 - ln(2) ≈ 0.30685 (30.685%)

   改进公式：理论成功率 = 1 - Σ(1/k) 对于 k 从 K+1 到 N

2. **动态规划**：

```
dp = [0]*(N+1)
dp[0] = 1
for i in range(1, N+1):
    dp[i] = sum(dp[j-1]/i for j in range(1, min(i, K)+1))
```

关键图表



数学解释

循环策略高效的原因

1. **排列的循环分解**：任何排列都可分解为不相交的循环
2. **关键洞察**：所有囚犯成功当且仅当最大循环长度≤K
3. **概率计算**：随机排列中最大循环长度≤N/2 的概率收敛于 1-ln(2)≈30.7%

随机策略低效的原因

- 单个囚犯成功概率：P_individual = K/N = 0.5
- 整体成功概率：P_all = (K/N)^N = (0.5)^100 ≈ 7.9 × 10^(-31)