中国科学技术大学 University of Science and Technology of China | 网络空间安全学院 School of Cyber Science and Technology

《密码学导论》课程大作业作品设计报告

作品题目：混沌置乱的循环阶分析

作者：何思宇

2025 年 6 月 7 日

基本信息表

作品题目：混沌置乱的循环阶分析

作品内容摘要：

以下是使用混沌映射构造置乱的一种常用方法：

1. 选定一个混沌映射。

2. 选定参数μ和初始值(即种子)x_0，迭代M轮得到x_M，M常取1000。继续迭代计算x_{M+1}~x_{M+N}

3. 将这N个数排序，以每个数的位置为置乱索引。例如，若x_i被排在第j位，则置乱中将第i个数移至第j位

编写一个程序对三种不同的混沌映射生成基于以上方法的置乱表，并评测该置乱表的循环情况。

关键词：

混沌映射，置乱，循环阶

1.作品功能与性能说明

主体是用 python 编写的程序，支持使用 logistic，singer，PWLCM 这三种混沌映射来生成置乱表并对其循环情况进行评测。

2.设计与实现方案

程序实现步骤总体上分为两部分：生成和测评。（均以 logistic 为例）

生成部分分为三块：

1. 混沌映射对应函数的实现；

2. 置乱表的生成；（伪代码如下）

```
定义 create_table(func, x0, n, a, N)
    检查 func 是否为支持的函数
    检查 a 是否为 None
    初始化序列 seq，x = x0
    循环 N 次：
        x = func(x, n, a)
        将 x 加入 seq
    对 seq 排序，返回排序索引 idx
```

3. 循环情况的评测；

```
定义 analyze_cycles(perm)
    初始化 visited 数组和 cycle_lengths 列表
    遍历 perm，查找所有循环圈长度
    统计每种长度的循环圈数量
    计算所有循环长度的最小公倍数 order
    返回循环圈长度统计和 order
```

测评部分则是两块：

1. 固定 N，改变初始值，算得平均阶以及阶按初值的分布

```
定义 average_order_vs_N(func, a, n, N_list, seeds_per_N)
        初始化 avg_orders 列表
        对每个 N in N_list:
            初始化 orders 列表
            对每个 seed in seeds_per_N:
                设置随机种子
                随机生成 x0
                生成置乱表 perm
                分析循环圈，得到 order
                将 order 加入 orders
            计算 orders 的均值，加入 avg_orders
        返回 avg_orders
```

2. 改变 N，绘得“平均阶-N”曲线

以上功能均以函数形式构建。

3. 结果展示：

此处以 logistic 为例进行展示：

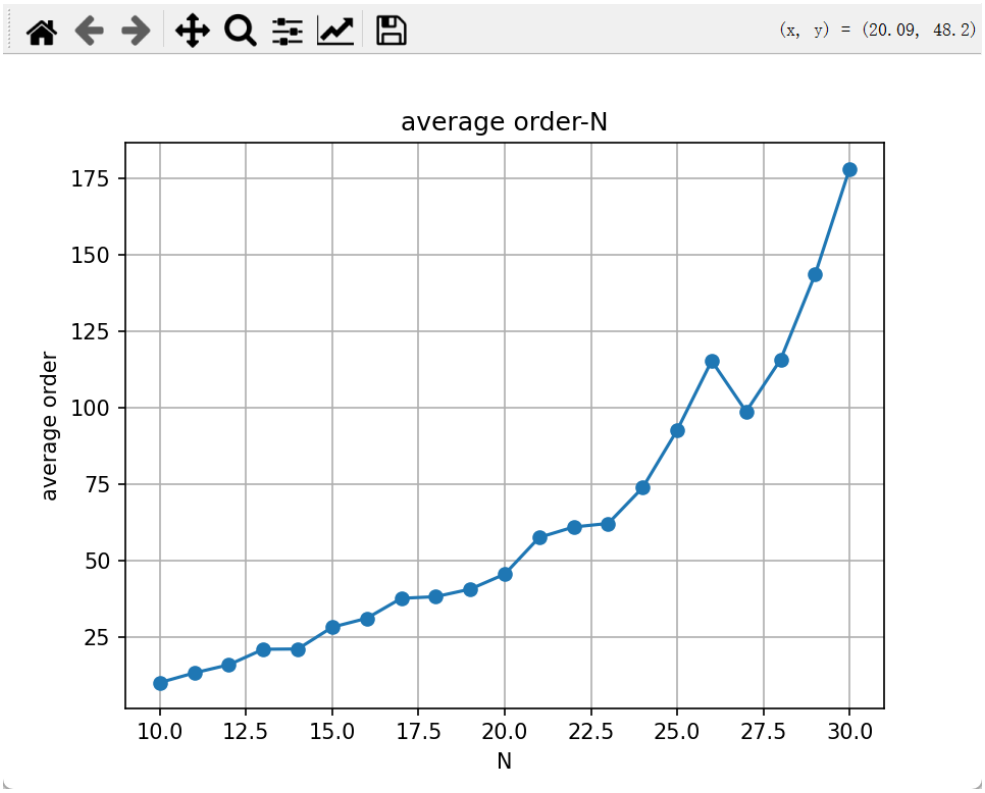
（1）生成置乱表：

```
PS C:\Users\hdy> & C:/Users/hdy/anaconda3/python.exe c:/Users/hdy/Desktop/create_table.py
请选择功能：
1. 生成置乱表
2. 计算平均阶并绘制平均阶-N曲线
请输入功能编号（1或2）：1
Enter the function name (logistic, singer, PWLCM): logistic
Enter the number of iterations (n): 1000
Enter the parameter 'a' (must be in range (3.57, 4)): 3.62
请输入置乱表长度N: 15
生成的置乱表：
[ 8 11  2  1 13  6 12  9  7 10  3 14  0  5  4]
置乱表的循环圈分析：
循环圈长度统计： Counter({14: 1, 1: 1})
置乱表的阶： 14
```

（2）计算平均阶并绘制平均阶-N 曲线：

```
PS C:\Users\hdy> & C:/Users/hdy/anaconda3/python.exe c:/Users/hdy/Desktop/create_table.py
请选择功能：
1. 生成置乱表
2. 计算平均阶并绘制平均阶-N曲线
请输入功能编号（1或2）： 2
Enter the function name (logistic, singer, PWLCM): logistic
Enter the number of iterations (n): 10000
Enter the parameter 'a' (must be in range (3.57, 4)): 3.77
N值及对应平均阶：
N=10, 平均阶=9.96
N=11, 平均阶=13.13
N=12, 平均阶=15.79
N=13, 平均阶=20.87
N=14, 平均阶=20.96
N=15, 平均阶=28.10
N=16, 平均阶=31.01
N=17, 平均阶=37.54
N=18, 平均阶=38.09
N=19, 平均阶=40.56
N=20, 平均阶=45.46
N=21, 平均阶=57.49
N=22, 平均阶=60.90
N=23, 平均阶=62.06
N=24, 平均阶=73.79
N=25, 平均阶=92.62
N=26, 平均阶=115.30
N=27, 平均阶=98.71
N=28, 平均阶=115.73
N=29, 平均阶=143.59
N=30, 平均阶=178.19
```

构建“平均阶-N”图像如下：



4. 结论

此次大作业难度不大，功能实现上并不困难，核心难度在于如何计算阶，此处稍微使用一部分代数学基础的知识即可。

https://github.com/hdy-hsy/crypto-homework.git