

# 中国科学技术大学

# 学士学位论文



## 密码学导论课程实践

### 题目三 BM 算法

作者姓名： 史昊霖

学科专业： 电子信息工程

导师姓名： 李卫海 副教授

完成时间： 二〇二一年五月二十六日



University of Science and Technology of China  
A dissertation for bachelor's degree



# **A experimental report of the Introduction to Cryptology course practice topic 3: BM algorithm**

Author: Shi Haolin

Speciality: Electronic Information Engineering

Supervisor: Asso. Prof. Weihai Li

Finished time: May 26, 2021



## 中文内容摘要

本科生需要手动将摘要置于目录后。

本文为密码学导论课程实践的题目三：**BM** 算法的实验报告。使用 **Python** 作为编程工具，以字符形式读取数据，按位转换为数字 0、1 进行操作，总体实现了对选定序列线性复杂度及其对应最小阶数的线性反馈移位寄存器的求解。验证了 **BM** 算法的时间复杂度。

**关键词：**中国科学技术大学；课程论文；密码学导论；**BM** 算法；实验报告

## Abstract

This paper is a experimental report of the Introduction to Cryptology course practice topic 3: BM algorithm. Python is used as a programming tool, it read data as characters, and convert them into digits 0 and 1 by bit. The linear complexity of the selected sequence and the linear feedback shift register of the minimum order corresponding to the sequence can be solved. The time complexity of BM algorithm is verified.

**Key Words:** University of Science and Technology of China (USTC); Course Paper; Introduction to Cryptology; Berlekamp-Massey; Experimental Report

## 致 谢

感谢李卫海老师在课程方面的认真讲解和指导帮助。

感谢助教们习题课的讲解以及在课余时间的问题解答。





## 目 录

中文内容摘要 . . . . .	I
英文内容摘要 . . . . .	II
第一章 实验程序介绍 . . . . .	2
第一节 算法流程图 . . . . .	2
一、BM 算法求解程序 . . . . .	2
第二节 运行效果图 . . . . .	3
一、求解题干中示例序列 1001101001101 . . . . .	3
二、求解 20Kbit 数据（数据来自预先生成的随机字节文件） . . . . .	3
第二章 实验结果分析 . . . . .	4
第一节 课程内简介 . . . . .	4
第二节 题目要求 . . . . .	4
第三节 密文测试 . . . . .	4
第四节 有效性测试 . . . . .	5
第三章 算法详细与结论 . . . . .	6
第一节 算法实现 . . . . .	6
第二节 结论说明 . . . . .	6
参考文献 . . . . .	8

## 第一章 实验程序介绍

### 第一节 算法流程图

#### 一、BM 算法求解程序

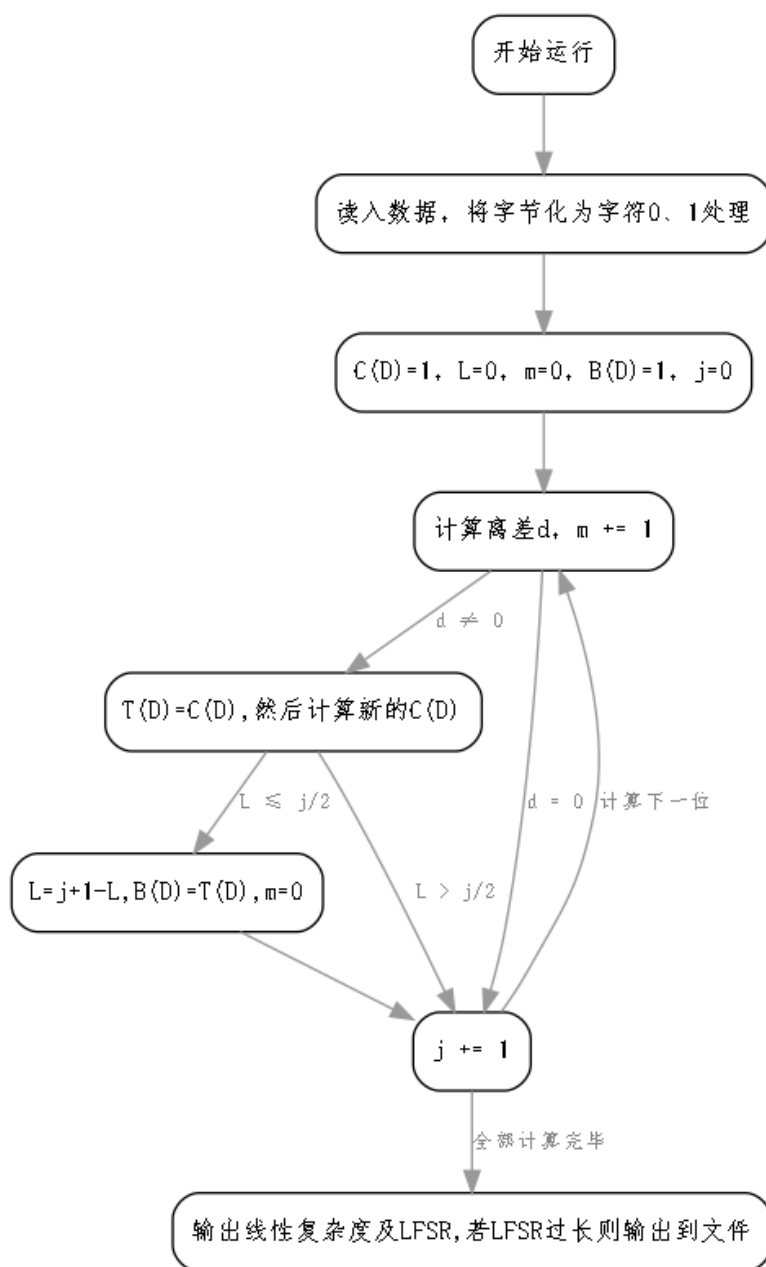
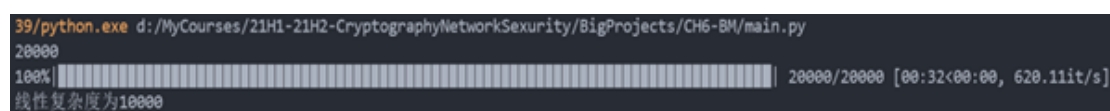


图 1.1 BM 算法-程序流程图

```
39/python.exe d:/MyCourses/21H1-21H2-CryptographyNetworkSecurity/BigProjects/CH6-BM/main.py  
1001101001101  
13  
100%|██████████████████████████████████████████████████████████████████████████████| 13/13 [00:00<?, ?it/s]  
线性复杂度为6  
The LFSR = <6, 1+D6>
```

## 二、求解 20Kbit 数据（数据来自预先生成的随机字节文件）



由于结果过长，将输出结果存到文本文件中。

图 1.4 输出的文本文件

## 第二章 实验结果分析

### 第一节 课程内简介

BM 算法全称 Berlekamp-Massey 算法。

用于求解最小阶数的线性反馈移位寄存器，使之输出的前  $n$  个比特与目标序列相同。

定义  $d = s_k + c_1 s_{k-1} + c_2 s_{k-2} + \cdots + c_L s_{k-L}$  为迭代到第  $k$  轮时的下一步离差。即第  $k-1$  轮迭代结果对  $s_k$  的预测与实际  $s_k$  的差。

运行时间为  $O(2^n)$  次比特操作。

设  $s$  的线性复杂度为  $L$ ，则以  $s$  的一个长度至少为  $2L$  的子序列为输入的 B-M 算法可以唯一确定生成  $s$  且长度为  $L$  的 LFSR。

### 第二节 题目要求

读入一个 0/1 串，串的长度在 10 比特到 1M 比特之间。

输出线性复杂度及相应的联结多项式。

以题目 2 的 AES 工具加密题目 1 中使用的书籍，在密文中任意截取若干长度为 1kb, 5kb, 10kb, 20kb 的子串，测试它们的线性复杂度。

### 第三节 密文测试

为便于对比，对于本次测试，采取同一个随机偏移量，故先使用 random 函数取一个范围  $[0, 134000]$  的字节偏移量，为 96995。

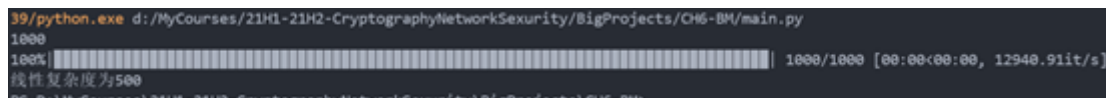


图 2.1 1Kbit 测试

线性复杂度：500 time 函数计时：130.28264045715332 ms



图 2.2 2Kbit 测试

线性复杂度：1000 time 函数计时：517.6577568054199 ms

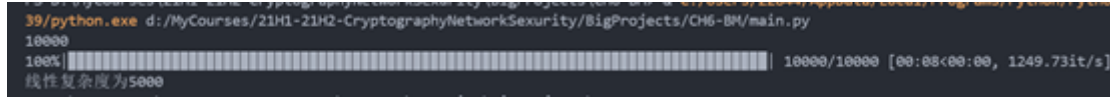


图 2.3 10Kbit 测试

线性复杂度：5000 time 函数计时：11528.244018554688 ms

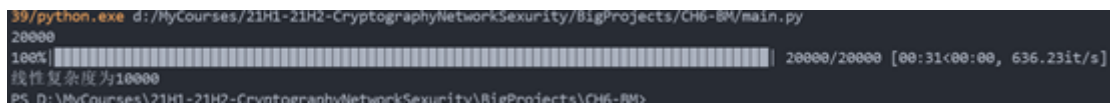


图 2.4 20Kbit 测试

线性复杂度：10000 time 函数计时：36218.80531311035 ms

## 第四节 有效性测试

采用 PPT 中示例序列、网络上搜索的测试序列进行测试，可以确认程序有效。

1. 序列：001101110（课堂 PPT 中示例）

线性复杂度为 5

The LFSR =  $\langle 5, 1+D^3+D^5 \rangle$

经调试、打印、对照，程序运算过程及运算中各项的值与教案中完全一致。

2. 序列：1001101001101（题干中示例）

线性复杂度为 6

The LFSR =  $\langle 6, 1+D^6 \rangle$

3. 序列：001011101（来源网络 [https://blog.csdn.net/qq\\_33877253/article/details/99622328](https://blog.csdn.net/qq_33877253/article/details/99622328)）

线性复杂度为 4

The LFSR =  $\langle 4, 1+D^1+D^2 \rangle$

4. 序列：10101111（来源网络）

线性复杂度为 4

The LFSR =  $\langle 4, 1+D^3+D^4 \rangle$

## 第三章 算法详细与结论

### 第一节 算法实现

程序使用单个 Python 文件实现。

实验程序源码地址：<https://github.com/lclichen/Crypt2021/tree/master/CH6-BM>  
算法3.1 主要描述计算过程。

### 第二节 结论说明

随着数据量的增大，每一位的计算量呈上升趋势，这既是由于  $L$  的增大使得步骤 2 中异或的次数增大，也由于  $C(D)$ 、 $B(D)$  的长度增加使得其单次乘法运算中需要的移位次数越来越多。

根据测试数据中的平均速度对比可以看出，位数增大  $N$  倍，单步计算的平均时间便增大  $N$  倍，同时因为有多少位便要进行多少次循环，步数也增大到原先的  $N$  倍。可以看出时间复杂度为  $O(n^2)$ 。

该算法主要内容为先确定一个多项式，然后逐位检验数据来修正多项式。

读取文件时首个字节在读取时会被去掉首位的 0，因此引入了简单的长度判断用于补 0。

在看 PPT 上算法内容时，由于没有进行放映，其拼写检查挡住了  $L \leq j/2$  中的等号，检查了很多次程序都看不出问题，之后尝试自行增加了等号后发现可以得出正确结果，然后还跑去问老师这个是不是 PPT 上写错了，实在是非常尴尬。

**算法 3.1** BM 算法**Data:** 序列文本**Result:** 线性复杂度及 LFSR

```

1  初始化类 BM;
2  初始化: 导入数据, 为各计算项赋初值;
3  while  $j < N$  do
4       $m += 1$ ;
5       $d = \text{data}[j]$ ;
6      if  $L \neq 0$  then
7           $i = 0$ ;
8          while  $1 \leq i \leq L+1$  do
9               $d = d \text{ XOR } (\text{data}[j - i] * C(D)[i])$ ;
10              $i += 1$ ;
11         end
12     else
13     end
14     if  $d \neq 0$  then
15          $T(D) = C(D)$ ;
16          $C(D) = C(D) \text{ XOR } B(D) * D^m$ ;
17         if  $L \leq j/2$  then
18              $L = j+1-L$ ;
19              $B(D) = T(D)$ ;
20              $m = 0$ ;
21         else
22         end
23     else
24     end
25      $j += 1$ ;
26 end

```

## 参 考 文 献