M序列拆圈猜想

唐小林　朱子恒　许楠楠　王岑
指导老师：林东岱

2017 年 5 月 14 日

1　M序列简述

M序列是q元r级非线性移位寄存器生成的具有最长周期的序列（含有q^r个不同状态），它与de Bruijn序列差了一个平移等价的条件。而我们常听说的m序列，是线性移位寄存器产生的具有最长周期的序列（含有q^r-1个不同状态）。像m,M序列这种长周期的序列在密码学中很有用处。但相对与m序列，M的个数要多得多，应用起来会有更多选择。m 序列可以由多项式方法生成，对于2元r级线性移位寄存器，可以生成$\Phi(2^r-1)/r$个序列，而M序列共有$2^{2^{r-1}-r}$ 个。

如图给出2元r级的m 序列与M 序列个数的对比。

类别	公式	1	2	3	4	5	6	7	8	9
m序列	$\Phi(2^r-1)/r$	1	1	2	2	6	6	18	16	48
M序列	$2^{2^{r-1}-r}$	1	1	2	16	2048	2^{26}	2^{57}	2^{121}	2^{248}

2　M序列拆圈原理

设

$$s=(a_1,\cdots,a_n),s*=(\overline{a_1},\cdots,a_n) \tag{1}$$

我们把s和s*叫做一对共轭状态或者一对共轭顶点。

Theorem 2.1. 设$f(x_1,x_2,\cdots,x_n)$是非奇异的n元开关函数.令$s=(a_1,a_2,\cdots,a_n),a_i\in\mathbb{F}_2$.令

$$f_s(x_1,x_2,\cdots,x_n)=x_2^{a_2}x_3^{a_3}\cdots x_n^{a_n} \tag{2}$$

$f+f_s$非奇异.

如果s和s^*属于以f为反馈逻辑的n级移位寄存器的状态图G_f 中不同的圈:

$$(s_0=s,s_1,\cdots,s_{k_1}),(t_0=s^*,t_1,\cdots,t_{k_2}) \tag{3}$$

那么以$f+f_s$为反馈逻辑的n级移位寄存器的状态图G_{f+f_s}可以将G_f的上述两个圈合并成一个圈

$$(s_0=s,t_1,\cdots,t_{k_2},t_0=s^*,s_1,\cdots,s_{k_1}) \tag{4}$$

并保持其余各圈不动而得到.

如果s和s^*属于G_f中同一个圈

$$(s_0,s_1,\cdots,s_k) \tag{5}$$

而

$$s=s_0,s^*=s_l(0<l\le k) \tag{6}$$

那么G_{f+f_s}可以将G_f的上述一个圈分成两个圈

$$(s_0=s,s_{l+1},s_{l+2},\cdots,s_k),(s_l=s^*,s_1,s_2,\cdots,s_{l-1}) \tag{7}$$

并保持其余各圈不动而得到.

Proof. $f_1 := f + f_s$.
如果$(b_2, b_3, \cdots, b_n) \neq (a_2, \cdots, a_n)$,那么

$$f_1(b_1, b_2, \cdots, b_n) = f(b_1, b_2, \cdots, b_n) \tag{8}$$

而

$$f_1(a_1, a_2, \cdots, a_n) = f(\overline{a_1}, a_2, \cdots, a_n) \tag{9}$$

$$f_1(\overline{a_1}, a_2, \cdots, a_n) = f(a_1, a_2, \cdots, a_n) \tag{10}$$

因此有

$$T_{f_1}(t) = T_f(t), \quad \forall t \neq s, s^* \tag{11}$$

而

$$T_{f_1}(s) = T_f(s^*), \quad T_{f_1}(s^*) = T_f(s) \tag{12}$$

由此可推出$G_(f + f_s)$与G_f的关系. □

由上述定理我们知道，对于M序列的拆圈问题，若一个圈中有两个共轭顶点，那么我们可以利用这对共轭对对该圈进行拆分。把M序列进行不断拆分，由共轭对的有限性，有限步后，能得到一些不能拆分的圈，那么这种分解得到的圈是否唯一？或者说是否有某种不变的量？我们并不知道，因而我们主要研究了2元4级和2元5级的序列，探究拆分M 序列后得到一些不能再拆分的圈是否具有一些不变量。

3 拆圈猜想

我们发现，有两类状态关系比较特殊。

一类为全为1的状态与全为0的状态（例如，对于2元3级移位寄存器，111与000两种状态；对于2 元4级移位寄存器，1010 与0101 两种状态）;另一类是0与1相隔，具有对称性的两种状态，我们称为为互生状态对，它们可以单独成圈，这个圈我们成为互生圈。（例如，对于2 元3级移位寄存器，101 与010 两种状态;对于2 元4 级移位寄存器，1010 与0101 两种状态 ; 依次类推。）

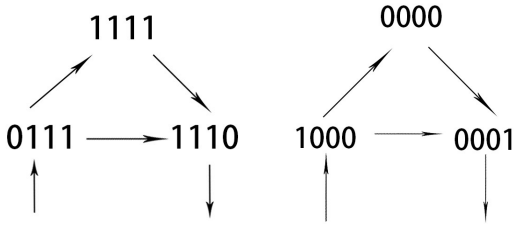


Figure 1: 含有全为1或者全为0

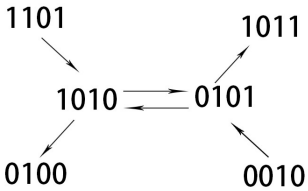


Figure 2: 含有互生状态对

对于第一类状态，全为1或全为0的状态可以拿出来单独成圈。 对于第二类状态，这两种状态要么在一起单独成圈，要么分开在不同的圈里。

由第二类状态我们可以给出拆分成最小圈的一个分类。一类是含有互生圈，另一类是不含有互生圈，具有对称性的两种状态在不同的两个圈中。对于这样一种分类再加上我们对2 元3,4,5级的分析，我们可以猜想在这样的分类下圈数可能具有某种统一量。我们猜想，对于分类的第二种情况，可能会形成4 个圈，互生状态对分开会导致其它共轭对也全分开了（除了含有全为0或全为1的共轭对）；2元r级M序列用共轭拆圈方法拆为一些不能拆分圈的圈的个数具有某种规律（级数$r \geq 3$）。

4 问题探究

对于2元1级的M序列共有1个，0，1单独成圈。

对于2元2级的M序列共有1个，拆分为00，11，互生圈。

对于2元3级的M序列共有2个，为

$(000) \to (001) \to (011) \to (111) \to (110) \to (101) \to (010) \to (100) \to (000)$ 和

$(000) \to (001) \to (010) \to (101) \to (011) \to (111) \to (110) \to (100) \to (000)$

每个M序列拆圈有两种拆法，但得到的圈个数都为4，一种拆法得到的小圈含有互生圈，另一种不含有。

对于2元4级的M序列共有16个，个数较多，情况较复杂，如下图（每个状态表示成了10进制如$1111_{(2)} = 15_{(10)}$，每一列为一个M序列，方向由下往上）

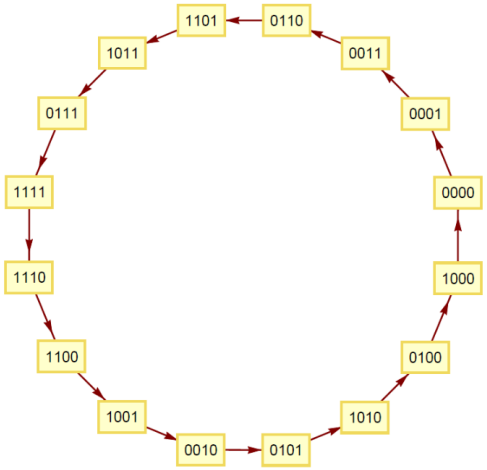
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
2	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
3	3	3	3	3	3	3	3	3	11	11	11	11	11	11	11	11
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
4	1	1	1	1	9	9	9	9	5	5	5	5	5	13	13	5
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
5	0	0	0	0	4	4	4	12	2	2	2	10	10	6	6	2
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
6	8	8	8	8	2	10	10	6	1	9	9	13	13	3	3	1
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
7	4	4	4	12	1	5	13	11	0	4	12	6	6	1	9	0
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
8	2	10	10	6	0	2	6	5	8	10	6	3	3	0	4	8
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
9	9	5	13	11	8	1	11	2	4	13	3	1	9	8	10	12
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
10	12	2	6	5	12	0	5	1	10	6	1	0	4	4	5	6
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
11	6	9	11	2	6	8	2	0	13	3	0	8	2	10	2	3
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
12	11	12	5	9	11	12	1	8	6	1	8	4	1	5	1	9
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
13	5	6	2	4	5	6	0	4	3	0	4	2	0	2	0	4
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
14	10	11	9	10	10	11	8	10	9	8	10	9	8	9	8	10
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
15	13	13	12	13	13	13	12	13	12	12	13	12	12	12	12	13
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
16	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14	14
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
17	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15

对于这16个M序列，我们用如下方法逐一拆分探究小圈的个数多少：

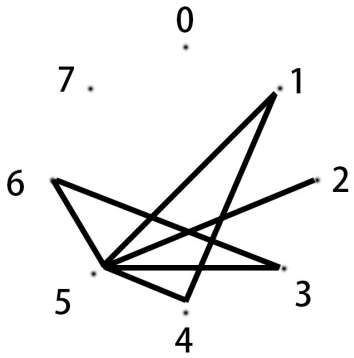
我们记共轭顶点{0000－1000}为0，{0001－1001}为1，{1010－0010}为2，{0011－1011}为3，{0100－1100}为4，{1101－0101}为5，{1110－0110}为6，{0111－1111}为7。（这里也可用十进制的共轭顶点，2元4级的话，它们相差8，2元n级相差$2^{(r-1)}$，如15与8为一对共轭顶点）

若拆掉一对共轭顶点**i**(这里讨论的是2元4级寄存器，故$0 \leq \mathbf{i} \leq 7$)的同时也把另一对共轭顶点$\mathbf{j} \neq \mathbf{i}$也拆散了,就说共轭顶点**j，i有关系**,并在共轭顶点图中把**j，i**相连。弄清各个点的关系后，进行拆圈，而拆圈次数被上述方法化为拆关系的次数，而圈个数与拆解关系的次数有关。

举个例子，这是一个M序列形成的圈:



共轭顶点关系就与如下图所示：



因为共轭顶点0，7中全1状态与全0状态可以单独成圈，故可先拆0，7，这已经拆分成了三个圈，状态数分别为1，1，14。接下来拆分的顺序不同，结果也就不同

a.可以先拆5，则由上图可以看出，其它共轭顶点全分散开来，不能再拆分，故此时可以拆分为4个圈。

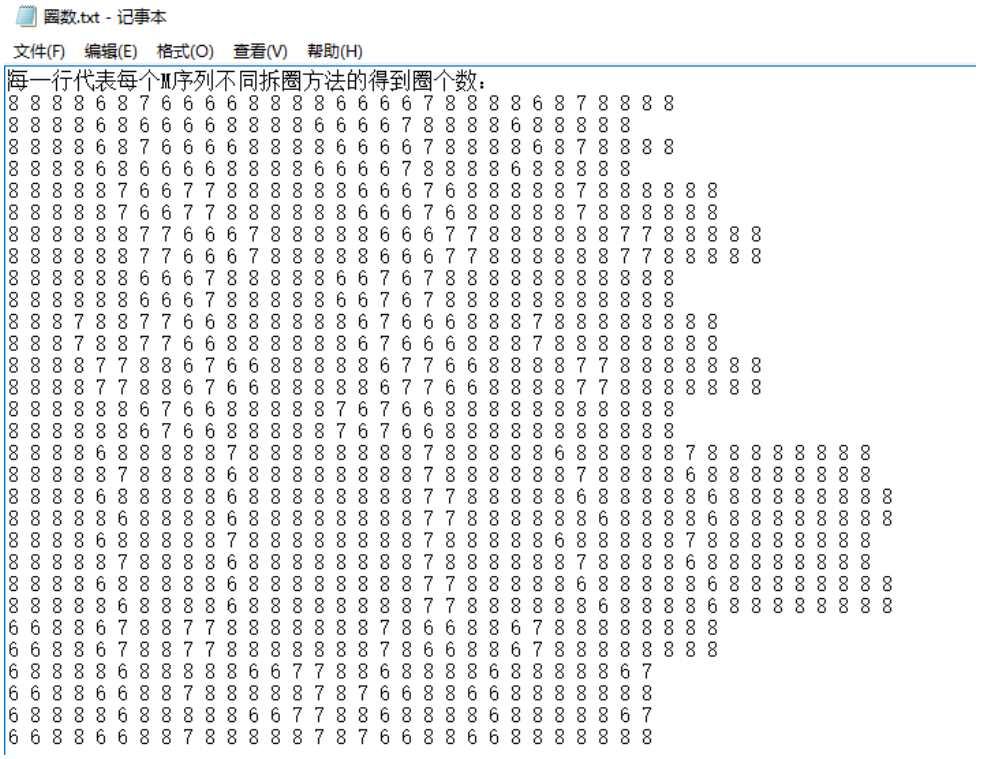
b.可以先拆分1，则由关系图可以看出，4，5被分散开来，便不用拆分4，5；接下来剩下2，3，6，可以由关系图看出3，6有关系，故接下来可以先拆分3，6其中一个，再拆分2（或者拆分2再拆分3或6，也是一个效果)。这样下来拆分三次，为六个圈。关系拆分顺序为$1 \rightarrow 3(6) \rightarrow 2$(也可为$1 \rightarrow 2 \rightarrow 3(6)$)

同理，可以拆分以2,3,4,6为头的关系，同样得到需要拆分3次，拆为6个圈。

……

我们将这16个M序列用上面的方法逐一试验，发现每个M序列拆分的结果要么是4，要么是6；但得到的圈数为4或6的序列是不同的，因此我们只能在圈数上去猜想是否有一些不变量。我们发现，圈个数为4 的不含有互生圈;圈个数为6的含有互生圈,这与我们之前的猜测些许符合。

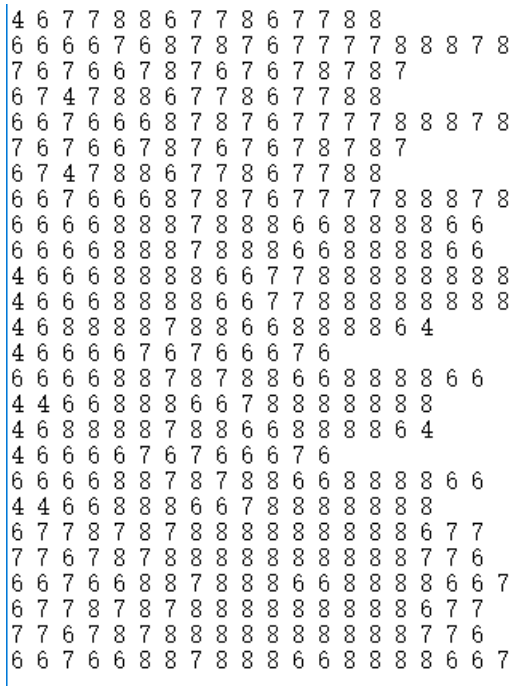
对于2元5阶的M序列共有2048个，数目较多，运用拆2元4级M序列的拆圈方法，用手算不太现实，但此算法比较机械，于是考虑写成程序，借助计算机来算出M序列拆圈的圈个数。首先借助计算机把所有M序列都遍历出来（详细过程见附件程序文件M_sequence.cpp）,接着用得到每一个M序列去计算不同拆圈方法下的不同圈数结果（因为每个M序列拆的共轭对不同，得到结果也不同,程序文件MseqGraph.cpp）。算圈数的程序文件MseqGraph.cpp算法思想和二元4阶M序列拆圈方法如出一辙，其中关系图用他们的邻接矩阵写出来，不过邻接矩阵的一般定义是对角线写0，这里方便后续操作将对角线改写为1；其中拆圈过程的完成用到了两个判断条件，一个意思是所有共轭对都被拆开了，一个意思是所选取的拆圈方式彼此不冲突，详细情况见文件。最终，我们得到的结果部分截图如下：


圈数.txt - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

每一行代表每个M序列不同拆圈方法的得到圈个数：

8 8 8 8 6 8 7 6 6 6 6 8 8 8 8 8 6 6 6 6 7 8 8 8 8 6 8 7 8 8 8 8
8 8 8 8 6 8 8 6 6 6 6 8 8 8 8 8 6 6 6 6 7 8 8 8 8 8 6 8 8 8 8 8
8 8 8 8 6 8 7 6 6 6 6 8 8 8 8 8 6 6 6 6 7 8 8 8 8 8 6 8 7 8 8 8 8
8 8 8 8 6 8 6 6 6 6 6 8 8 8 8 8 6 6 6 6 7 8 8 8 8 8 6 8 8 8 8 8 8
8 8 8 8 8 7 6 6 7 7 8 8 8 8 8 8 8 6 6 6 7 6 8 8 8 8 8 8 7 8 8 8 8 8 8
8 8 8 8 8 8 7 6 6 7 7 8 8 8 8 8 8 8 6 6 6 7 6 8 8 8 8 8 8 8 7 8 8 8 8 8 8
8 8 8 8 8 8 8 7 7 6 6 6 7 8 8 8 8 8 8 6 6 6 7 7 8 8 8 8 8 8 8 7 7 8 8 8 8 8 8
8 8 8 8 8 8 8 7 7 6 6 6 7 8 8 8 8 8 8 6 6 6 7 7 8 8 8 8 8 8 8 7 7 8 8 8 8 8 8
8 8 8 8 8 8 8 6 6 6 7 8 8 8 8 8 8 8 6 6 7 6 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
8 8 8 8 8 8 8 6 6 6 7 8 8 8 8 8 8 8 6 6 7 6 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
8 8 8 7 8 8 8 7 7 6 6 8 8 8 8 8 8 8 8 6 7 6 6 6 8 8 8 8 7 8 8 8 8 8 8 8 8 8
8 8 8 7 8 8 8 7 7 6 6 8 8 8 8 8 8 8 8 6 7 6 6 6 8 8 8 8 7 8 8 8 8 8 8 8 8 8
8 8 8 8 7 7 8 8 8 6 7 6 6 8 8 8 8 8 8 8 6 7 7 6 6 8 8 8 8 8 7 7 8 8 8 8 8 8 8
8 8 8 8 7 7 8 8 8 6 7 6 6 8 8 8 8 8 8 8 6 7 7 6 6 8 8 8 8 8 7 7 8 8 8 8 8 8 8
8 8 8 8 8 8 8 6 7 6 6 8 8 8 8 8 8 7 6 7 6 6 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
8 8 8 8 8 8 8 6 7 6 6 8 8 8 8 8 8 7 6 7 6 6 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
8 8 8 8 8 6 8 8 8 8 8 8 7 8 8 8 8 8 8 8 8 7 8 8 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8
8 8 8 8 8 7 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8 7 8 8 8 8 8 8 8 7 8 8 8 8 8 6 8 8 8 8 8 8 8 8
8 8 8 8 6 8 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8 7 7 8 8 8 8 8 8 6 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8
8 8 8 8 8 6 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8 7 7 8 8 8 8 8 8 6 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8
8 8 8 8 6 8 8 8 8 8 8 7 8 8 8 8 8 8 8 8 8 7 8 8 8 8 8 8 8 6 8 8 8 8 8 7 8 8 8 8 8 8 8 8
8 8 8 8 8 7 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8 7 8 8 8 8 8 8 8 7 8 8 8 8 8 6 8 8 8 8 8 8 8 8
8 8 8 8 6 8 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8 7 7 8 8 8 8 8 8 6 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8
8 8 8 8 8 6 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8 7 7 8 8 8 8 8 8 6 8 8 8 8 8 6 8 8 8 8 8 8 8 8 8
6 6 8 8 8 6 7 8 8 8 7 7 8 8 8 8 8 8 8 8 8 7 8 6 6 8 8 8 6 7 8 8 8 8 8 8 8 8
6 6 8 8 8 6 7 8 8 8 7 7 8 8 8 8 8 8 8 8 8 7 8 6 6 8 8 8 6 7 8 8 8 8 8 8 8 8
6 8 8 8 8 8 6 8 8 8 8 8 8 6 6 7 7 8 8 8 6 8 8 8 8 8 6 8 8 8 8 8 8 6 7
6 6 8 8 8 6 6 8 8 7 8 8 8 8 8 8 7 8 7 6 6 8 8 8 6 6 8 8 8 8 8 8 8 8
6 8 8 8 8 8 6 8 8 8 8 8 8 6 6 7 7 8 8 8 6 8 8 8 8 8 6 8 8 8 8 8 8 8 6 7
6 6 8 8 8 6 6 8 8 7 8 8 8 8 8 8 7 8 7 6 6 8 8 8 6 6 8 8 8 8 8 8 8 8


4 6 7 7 8 8 6 7 7 8 6 7 7 8 8
6 6 6 6 7 6 8 7 8 7 6 7 7 7 7 8 8 8 7 8
7 6 7 6 6 7 8 7 6 7 6 7 8 7 8 7
6 7 4 7 8 8 6 7 7 8 6 7 7 8 8
6 6 7 6 6 6 8 7 8 7 6 7 7 7 7 8 8 8 7 8
7 6 7 6 6 7 8 7 6 7 6 7 8 7 8 7
6 7 4 7 8 8 6 7 7 8 6 7 7 8 8
6 6 7 6 6 6 8 7 8 7 6 7 7 7 7 8 8 8 7 8
6 6 6 6 8 8 8 7 8 8 8 6 6 8 8 8 8 6 6
6 6 6 6 8 8 8 7 8 8 8 6 6 8 8 8 8 6 6
4 6 6 6 8 8 8 8 8 6 6 7 7 8 8 8 8 8 8 8 8
4 6 6 6 8 8 8 8 8 6 6 7 7 8 8 8 8 8 8 8 8
4 6 8 8 8 8 8 7 8 8 6 6 8 8 8 8 6 4
4 6 6 6 6 7 6 7 6 6 6 7 6
6 6 6 6 8 8 7 8 7 8 8 6 6 8 8 8 8 6 6
4 4 6 6 8 8 8 8 6 6 7 8 8 8 8 8 8 8
4 6 8 8 8 8 8 7 8 8 6 6 8 8 8 8 6 4
4 6 6 6 6 7 6 7 6 6 6 7 6
6 6 6 6 8 8 8 7 8 7 8 8 6 6 8 8 8 8 6 6
4 4 6 6 8 8 8 8 6 6 7 8 8 8 8 8 8 8
6 7 7 8 7 8 7 8 8 8 8 8 8 8 8 8 6 7 7
7 7 6 7 8 7 8 8 8 8 8 8 8 8 8 8 7 7 6
6 6 7 6 6 8 8 7 8 8 8 8 6 6 8 8 8 8 6 6 7
6 7 7 8 7 8 7 8 8 8 8 8 8 8 8 8 6 7 7
7 7 6 7 8 7 8 8 8 8 8 8 8 8 8 8 7 7 6
6 6 7 6 6 8 8 7 8 8 8 8 6 6 8 8 8 8 6 6 7

从结果可以看出M序列可以拆分为圈数情况有4,6,7,8。可以拆出圈数为4,6,7,8占2048个M序列的情况如下表：

拆出圈的个数	4	6	7	8	同时有7,8
M序列（共2048个）	192	1984	2016	2016	2048

故由以上结果，4,6,7,8分布不固定，圈数为4的数量较少，实在看不出有什么简单的不变量关系。对于我们的猜想，也不是每个M 序列都能拆出圈为4。有的M序列能拆出其中不含有互生圈的4个圈，但也能拆出不含互生圈的7 个圈，没有什么漂亮的结果，比较失望。或许我们由之前的探究改进猜想，级数大于等于3 时，每一级都存在一个M 序列可以拆出4 个圈。这还需要一些6 级等多级的M 序列去探究，但5 级以后的M 序列非常多，有相当大的困难。这样的猜想结果似乎意义不大，还不如在探究5 级以上的M 序列拆分圈上多花时间，看结果有没有什么规律，再来猜想，毕竟例子太少。这个问题的结果感觉比较复杂。

5 总结

通过探究一些特殊状态：全1与全0状态，互生状态对，发现，M序列拆圈一定能拆出全1，全0状态单独成圈。而对于互生状态对，我们猜想，当级数$r \geq 3$时,分开互生状态对拆的圈个数可能会为4，故存在一个M序列可以拆分为4个圈;而不分开互生状态对的有互生圈的，拆圈后个数可能有一些不变量关系。

为了验证猜想，我们主要拆圈了2元4级M序列，给出了一个得到圈个数的简单方法，又用此方法编写了一个程序进一步验证2元5级M序列。从目前结果来看，并没有什么漂亮的简单不变量关系，但对于2元r（$r \geq 3$）级M序列总是存在一个M序列可以拆分为4个圈（有些牵强）。而想对更高级数的M 序列拆分，数目多，周期长，拆分较困难。

参考文献

[1] 万哲先. 代数与编码（第三版）,高等教育出版社,2007.6

创新实践训练计划总结

通过中科院科创训练计划项目的这一年里，我们小组四人在中科院信工所林东岱老师，首都师范大学的葛根年老师，张俊老师，张一炜老师的指导下，系统学习了有限域，线性移位寄存器，非线性移位寄存器的部分理论知识，每周都开展讨论班讨论，虽然没有什么大成果，但学到了很多东西，也了解了做研究的一个大致过程。感谢中科院有个这么一个中科院训练计划，也更感谢老师的悉心教导与每周讨论班的陪伴。谢谢每位伙伴的辛勤付出与积极讨论，我们从各自的身上学到了很多东西。我们共做了两本总结，除了本论文研究M序列拆圈的总结外（下半年），还有一本是我们学习研究线性移位寄存器的读书笔记总结（上半年）。

导师意见