文章编号:1673-9469(2012)04-0079-04

产生m子序列的一种实用算法

方俊初[1],吕 虹[2],张爱雪[1]
(1.安徽工程大学电气工程学院,安徽 芜湖 241000 2.安徽建筑工业学院 电子与信息学院,安徽 合肥 230022)

摘要:在基于m序列形成同级m子序列的过程中,核心问题是判断两对共轭状态的连线在"圈"内是否相交。本文将"圈"中各状态和整数建立了映射关系,从而给每一对共轭状态一个编码,通过该编码可以系统地判断任意两对共轭状态的连线在圈内是否相交。利用这一方法,可以方便地统计出子序列的数目,并能编程实现所有子序列的输出。

关键词:m子序列;映射;编码
中图分类号:TN919; TN431 **文献标识码**:A

An effective arithmetic to produce m-subsequence

FANG Jun-chu, LV Hong, ZHANG Ai-xue
(1. School of Electrical Engineering, Anhui Polytechnic University, Anhui Wuhu 241000, China; 2. School of Electronic and Information Engineering, Anhui University of Architecture, Anhui Hefei 230022, China)

Abstract: In the process of same level m-subsequence formed based on m-subsequence, the main question is to judge whether the lines of two sets of conjugate states will be intercrossed in the circle. In this passage, all situations and digital in this circle are built a relationship of reflection, and then each set of conjugate state has a code. Through this code, whether the lines of any two sets of conjugate states intercrossed in the circle will be judged by system. In this way, the number of subsequence can be calculated conveniently, and its program made as well. Then, the output of all the subsequences will be realized. .

Key words: m-subsequence; mapping; coding

若n级移位寄存器的特征多项式为n阶本原多项式,则该移位寄存器从任意一个非"0"状态出发所形成的状态图是一个包含2^n-1个状态的大"圈",如图1所示。由它产生的GF(2)序列的周期也是2^n-1,称为最大长度序列,即m序列,是一类常用的伪随机序列。在这2^n-1个状态中,某状态S的下一个状态称为S的"后继",而S的前一个状态称为S的"前驱"。而形如$S=(\alpha_0,\alpha_1,\alpha_2,\cdots\alpha_{n-1})$,$S^*=(\alpha_0,\alpha_1,\alpha_2,\cdots\overline{\alpha}_{n-1})$则称为一对共轭状态。

若两对共轭状态的连线在"圈"内存在交点,则交换它们的后继仍能形成一个长度为2^n-1的大"圈",与这个新的状态转换较图对应的序列称为原序列的子序列[1-2]。子序列形成过程如图2所示,图中S_1和S_p是一对共轭状态,S_k和S_q是另一对共轭状态。实线箭头表示原有的状态转换方向,虚线箭头表示新的状态转换方向。

在图2所示过程中,两对共轭状态的连线(图中实线所示)在圈内相交是这一转换过程成功的关键。但是单从状态转换的角度来判断哪两对共轭状态在"圈"内有交点存在很大的困难,因而文献[1]只在每一级中确定了一个子序列的生成方法。

实际上,在n阶本原多项式$f(x)$决定的线性移位寄存器中共有2^n-1个状态,因为00…01在圈内没有共轭状态,其余状态将两两互为共轭,因此共有$\frac{2^n-2}{2}$对共轭状态,这些共轭状态中符合上述交换条件的可能有很多,如何能快速准确的找

收稿日期:2012-09-10
基金项目:国家自然科学基金(No.61071001);安徽工程大学青年基金(2008YQ031zd)
作者简介:方俊初(1974-),男,安徽六安人,硕士学位,讲师,从事数字信号处理、电子技术应用方面的研究。

到这些满足条件的共轭状态呢? 又如何通过它们简便而又快速地实现这些 m 子序列呢? 本文利用数学变换思想,建立了一种"映射",将 $GF(2)$ 域中"状态"映射到整数域中,共轭状态可以用编码表示,通过该编码很容易判断出两对共轭状态在圈内有没有交点。

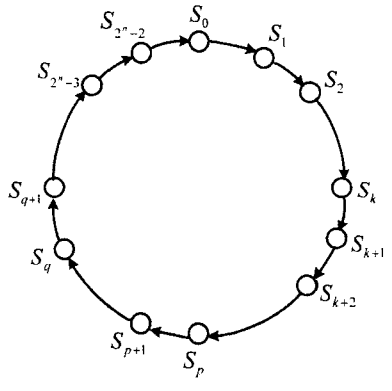

图 1 m序列状态图

Fig. 1 State diagram of m-sequence

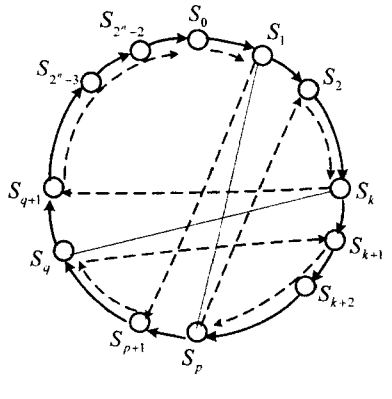

图2 m子序列形成过程

Fig. 2 Rebuilding of m-subsequence

1 建立一种映射关系

表 1 是以 $GF(2)$ 上以本原多项式 $f(x)=1+x^2+x^5$ 为反馈函数生成的移位寄存器的状态表[3-4]。

定义 1:在给定本原多项式的基础上,从任意一个初始状态 S_0 出发,按照状态出现的先后顺序将这些状态依次记为 S_0、S_1、$S_2\cdots S_{2^n-2}$ 下标 0、1、2 …(2^n-2)是该状态的序号。

某个状态的前驱与后继仅与反馈函数相关而与初始状态无关。因而可以得出:两对共轭状态在圈内的连线是否会相交,与初始状态的选择无关[5]。

这样 2^n-1 个"状态"就和 0、1、2…2^n-2 共 (2^n-1) 个整数建立了"映射"关系。编程时,假设这些状态以数组的形式存在,那么知道了这个"序号"就可以找到("引用")该"状态",已知"状态"也可以判断其在"圈"中的位置。这为下面研究共轭状态的相互关系提供了方便。

2 共轭状态的表示方法

定义 2:某一状态为 $S_k=(\alpha_0,\alpha_1,\alpha_2,\cdots\alpha_{n-1})$,设其共轭状态为 $S_p=\overline{(\alpha_0,\alpha_1,\alpha_2,\cdots\alpha_{n-1})}$,将它们的连线画在圈内,若 $k<p$ 则将这一对共轭状态记为[k,p],若 $k>p$ 则将这一对共轭状态记为[p,k],称为这一对共轭状态的"坐标"。

共轭状态的"坐标"包含了这一对共轭状态在圈中的位置信息。[k,p]中 k 称为起点,p 为终点,按上述定义,起点值一定比终点值要小,即要求 $0\le k<p\le 2^n-2$,举个例子说,表 1 中 S_3 和 S_5 是一对共轭状态,将其坐标记为[3,5],而不是[5,3]。这样规定有两个目的,一是确定了每一对共轭状态只有唯一的一个坐标了;二是方便用"遍历"法生成共轭状态表。有了共轭状态表,就可以通过这些坐标来研究共轭状态在圈中的相互关系了。表 2 是根据表 1 按上述定义 2 编程得到的共轭状态表。由表 2 可见共轭状态表的特点一是起点是按从小到的顺序排列的,二是每一对共轭状态在表中只出现一次。

图 3 画出了表 2 中共轭状态在圈中的连线。可见,这些共轭状态的连线在圈内的有的存在交点,有的不存在交点。有交点就可以产生新的序列,问题是在没有画出这个"圈"的情况下,如何才能确定两对共轭状态在圈内有无交点呢?

表 1 状态表实例

Tab. 1 An example of state table

序号	0	1	2	3	4	5	6	7	8	9	10
状态	00001	10000	01000	10100	01010	10101	11010	11101	01110	10111	11011
序号	11	12	13	14	15	16	17	18	19	20	21
状态	01101	00110	00011	10001	11000	11100	11110	11111	01111	00111	10011
序号	22	23	24	25	26	27	28	29	30		
状态	11001	01100	10110	01011	00101	10010	01001	00100	00010		

表 2 共轭状态表

Tab. 2 Conjugated state table

C0	[	1	14]	C5	[	7	16]	C10	[13	30]
C1	[	2	28]	C6	[	8	19]	C11	[15	22]
C2	[	3	5]	C7	[	9	24]	C12	[17	18]
C3	[	4	25]	C8	[	11	23]	C13	[21	27]
C4	[	6	10]	C9	[	12	20]	C14	[26	29]

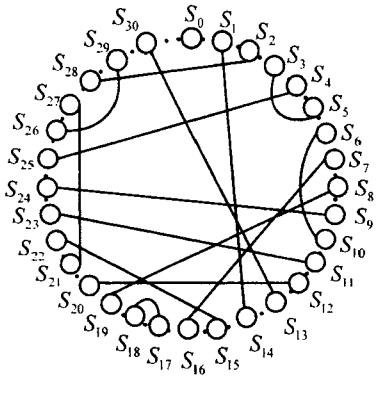

图3 共轭状态在“圈”内的连线(1)

Fig. 3 Ligature of conjugated state in the “circle” (1)

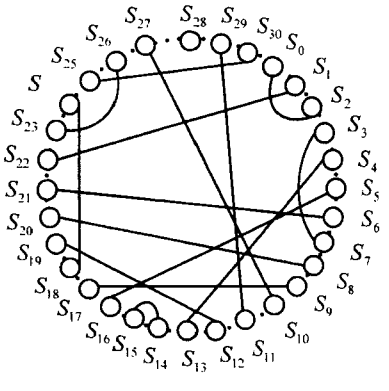

图4 共轭状态在“圈”内的连线 (2)

Fig. 4 Ligature of conjugated state in the “circle” (2)

3 子序列生成条件的简便判断

观察图 3 中这些连线的相互关系,并结合表 2,可以得到下面的定理。

定理 1:若[k,p]是一对共轭状态,[m,n]是另一对共轭状态,将起点坐标较小的这一对共轭状态称第一对共轭状态,另一对称为第二对共轭状态,这里假设[k,p]是第一对共轭状态,若 $k<m<p<n$,则这两对共轭状态的连线在圈内一定有交点。

证明:第一对共轭状态[k,p]的连线将“圈”一分为二,从起点出发,顺时针方向的一侧称为“右侧”,逆时针方向的一侧称为“左侧”。$k<m<p$ 即第二对共轭状态的起点位于第一对共轭状态的“右侧”,而 $n>p$,根据定义 1,则容易知道 $n>k$,即第二对共轭状态的终点一定位于第一对共轭状态的“左侧”,因而这两对共轭状态的连线在“圈”内必有交点,证毕。

定理 1 用数学语言将原来只能用文字表述的交换条件“数学化”了,这种数字化的结果是很方便编程统计出交点的个数,也能方便地判断两对共轭状态是否满足交换条件。例如上图中的共轭状态[1,14]和[2,28]在圈内有交点,而[1,14]和[6,10]在圈内就没有交点。

应当指出的是,在反馈函数确定的情况下,初始状态不同,则各状态在圈中的序号都将改变,共轭状态的坐标也就改变,如图 4 所示。用上述定义 1 规定的共轭状态的表示方法仍能反映各共轭状态在圈中的相互关系,如图 3 中[1,14]和[2,28]对应在图 4 就是[11,29]和[25,30],仍可判断它们的连线在圈内有交点。而图 3 中的[1,14]和[6,10]在图 4 对应的坐标是[11,29]和[3,7]仍能判断它们在圈内没有交点。

总之,共轭状态在圈内是否有交点与初始状态无关,从任一个初始状态出发进行编号,并按定义 1 生成的共轭状态表都能准确反映共轭状态在“圈”内的相互关系。

4 应用

4.1 子序列个数的统计

从一个已知的 m 序列的状态图,可以得到多少个子序列呢? 显然,这取决于共轭状态的连线在圈内有多少个交点,有了共轭状态表,只需按定理 1 统计出交点的个数,就是能生成的子序列的个数了。

设共轭状态表存储在二维数组 $C[RR][2]$ 中,其中 $RR=\frac{2^n-2}{2}$,是共轭状态的对数。用 C 语言编写下面的程序即可统计出交点的个数。

```
for(i=0;i<RR;i++)
    for(j=i+1;j<RR;j++)
    if(c[i][0]<c[j][0]&&c[i][1]
      <c[j][1]&&c[j][0]<c[i][1])
      num=num+1;
printf("子序列个数为: %3d\n",num);
```

实验结果:经过实验发现,相同级数的本原多项式对应的线性移位寄存器的共轭状态在圈内的交点数是相同的,例如前面讲的以 $f(x)=1+x^2+$

x^5 为反馈函数的移位寄存器,其共轭状态在圈内的交点数为35,若反馈函数换为 $f(x)=1+x^3+x^5$,其共轭状态在圈内的交点数也是35,其他的五阶本原多项式的情况也是如此。表3给出的是通过实验得到的级数为5-10级的线性移位寄存器共轭状态在圈内的交点数(子序列个数)。

表3 共轭状态交点数

Tab.3 The number of intersect points

级数n	5	6	7	8	9	10
周期	31	63	127	255	511	1 023
交点数	35	155	651	2 667	10 795	43 435

我们知道,n 阶线性移位寄存器所能产生的 m 序列(由不同的本原多项式产生)的数目远小于 m 序列的周期 N,但从表3可以看出,一个本原多项式衍生出的子序列的数目则远远大于它的周期,这些子序列保留了 m 序列的优秀伪随机特征。

4.2 子序列的生成

每一对共轭状态都对应一条连线,不妨将小项 $x_0^{a_0}x_1^{a_1}\cdots x_{n-2}^{a_{n-2}}$ 称为它们连线的特征小项[6]。其中 $x_0^{a_0}x_1^{a_1}\cdots x_{n-2}^{a_{n-2}}=$

$$\begin{cases}0(x_0x_1\cdots x_{n-2})\neq(a_0a_1\cdots a_{n-2})\\1(x_0x_1\cdots x_{n-2})=(a_0a_1\cdots a_{n-2})\end{cases}$$

下面以定理的形式给出生成 m 子序列的一般方法。

定理2:若[k,p]是一对共轭状态,[m,n]是另一对共轭状态,若 $k<m<p<n$,或 $m<k<n<p$,则在移位寄存器的反馈函数中加上(模2加)第 m 项和第 k 项的特征小项后,该移位寄存器的状态图将形成一个新的长度为 2^n-1 的大圈。对应此大圈将产生一个新的不同于原来的序列。

证明:$k<m<p<n$,或 $m<k<n<p$ 保证了两对共轭状态在"圈"必有交点。移位寄存器的反馈函数中加上(模2加)第 m 项和第 k 项的特征小项,意味着在上述四个状态后面的反馈函数值取反,即交换了次状态,重新形成了新的大"圈",证毕。

简单地讲,交换共轭状态的后续状态就是在状态生成过程中,遇到共轭状态中的一个就将原反馈函数值取反[7-8]。

程序实现的思路是:指定反馈移位寄存器的级数,输入相应的本原多项式的系数,从任一初始状态出发,将全部状态生成并按顺序储存,根据共轭状态的特点自动生成共轭状态表,给定一对共轭状态,判断是否符合交换条件,若符合交换条件,再次生成状态表时在该状态的下一个状态输出前通过异或"1"改变原有的反馈函数值。这中间可以用序号引用该状态[9-10]。

本文根据上述思想设计了程序,只要改变几个常数就可以生成任意级数本原多项式的全部 m 子序列或其中一个子序列,运行非常方便,非常适用在需要大量伪随机序列的场合。

5 结论

1)本文解决了"如何判断两对共轭状态在圈内有交点"这一形成子序列的关键问题。解决问题的思路简洁,非常便于程序实现。

2)本文介绍的方法可以在本原多项式的基础上可方便地产生大量的 m 子序列,为进一步研究这些子序列的性能奠定了基础,也必将进一步扩展伪随机序列在测量、通信、流密码等领域中的广泛应用。

参考文献:

[1] 吕虹,段颖妮,管必聪,等. 第一类m子序列的构造[J]. 电子学报,2007(10):2029-2032.

[2] LV HONG. Design and implementation of a maximal length nonlinear pseudorandom sequence[C]. Proceedings of the 2009 International Conference on Computer and Communications Security. 2009:12-16.

[3] 谢深泉. De Bruijn序列间的映射及升级算法[J]. 计算机工程与应用,2007(22):12-15.

[4] 谢深泉. De Bruijn序列查寻表标签的定值构造法[J]. 计算机工程与应用,2008(19):37-40.

[5] LAN JINGJING, GOH WANG LING, KONG ZHI HUI, et al. A random number generator for low power cryptographic application[C]. Proceedings of the 2010 International Soc. Design Conference, ISOCC 2010, 328-331

[6] 林智慧,陈绥阳,王元一. m序列及其在通信中的应用[J]. 现代电子技术,2009(09):49-52.

[7] 苏绍璟,伍文君,黄芝平,等. 含错m序列本原多项式的高阶统计测定算法[J]. 兵工学报,2010(12):1593-1597.

[8] 张晓林,佟婧,李佑虎. 高阶统计分析的m序列检测新方法[J]. 哈尔滨工程大学学报,2010(03):386-390.

[9] 贾银洁,赵丽娟,许鹏飞. 扩频系统中伪随机码发生器的实现[J]. 现代计算机(专业版),2008(05):72-74.

[10] 熊睿佳,胡万利. 伪随机m序列特性及C语言实现[J]. 工程地球物理学报,2011(01):110-112.

(责任编辑 刘存英)