DOI:10.15918/j.tbit1001-0645.1990.s3.016

北京理工大学学报
第10卷 第4期(Ⅱ) Journal of Beijing Institute of Technology Vol.10 No.4(Ⅱ) 1990

m序列优选对及平衡Gold码序列

辛肖明 陈 琼

(电子工程系)

摘 要 目前，在扩频选址通信中，国内外应用较广泛的Gold码，其构成采用的m序列优选对并非是最优的，有的还不是平衡的．本文提供一种求m序列优选对及其构成平衡Gold码的计算机算法．实际表明，这种方法使计算量大为减小，计算速度提高，计算结果准确，因而是一种较好的方法．

关键词 M序列／m序列．Gold码，平衡Gold码．

1 基本概念

1.1 m序列族及其某些特点

m序列即为n级移位寄存器通过线性反馈得到的周期为2^n-1的最长序列．

码序列的自相关函数可以用码字的一致数减去不一致数而得到的图表和曲线来表示．对于m序列，除了零位移和在同步条件外，净相关值为-1，而在零位移时净相关值为2^n-1．在位移为零和± 1比特之间的区域，相关值是线性增加的，即m序列的自相关函数呈现如图1所示的三角形．

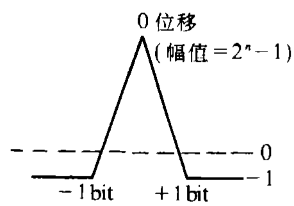

图1 m序列自相关函数曲线

码序列的互相关函数可以象自相关函数一样，以一致数减去不一致数而得到的图表和曲线来表示．

在所有m序列中，互相关函数最大值的绝对值$|\rho_{\max}|$中最小的一对m序列称为m序列优选对．它们的互相关函数满足

$$|\rho_{\max}| \leqslant D = \begin{cases} 2^{(n+1)/2}+1 & \text{当} n \text{为奇数时} \\ 2^{(n+2)/2}+1 & \text{当} n \text{为偶数，但不为4的倍数} \end{cases} \tag{1}$$

有一些m序列对是具有3值互相关函数：-1、$-t(n)$和$t(n)-2$，$t(n)=1+2^{[(n+1)/2]}$（[ ]表示取整）．这样的m序列对称为理想m序列对．对于所有的m序列优选对都是理想m序列对，只不过3值出现的位置和概率不同．

m序列优选对，虽然有较小的互相关幅度，但其地址数太少[1,2]，不能满足多址的要求．

1.2 Gold码序列及其特点

Gold码是由同步时钟控制的一对m序列逐比特模二加得到的，如图2所示．

1989年11月30日收稿

由于两个m 序列发生器的相对相移量可以有 2^n-1 种，所以 Gold 码序列发生器能产生 2^n-1 个长度是 2^n-1 比特的 Gold 码序列．再加上两个基本的 m 序列，共有 2^n+1 个 Gold 序列．Gold 序列族 $G(a,b)$ 的全部序列为

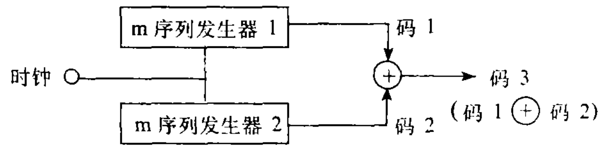

图 2 Gold 码序列发生器

$$G(a,b)=\{a,b,a\oplus b,a\oplus bT,\cdots,a\oplus bT^{2^n-2}\} \quad (2)$$

简记为 G 序列族．G 族中 a 及 b 的自相关函数为 2 值相关函数．$a\oplus bT^i$ 的自相关函数由 a 和 b 间的互相关函数决定．可以证明[3]，$a\oplus bT^i$ 的自相关函数与 a 及 b 间的互相关函数相等，也为 3 值相关函数．

$a\oplus bT^{i_1}$，$a\oplus bT^{i_2}$ 间的互相关函数也与 a，b 间的互相关函数相同．即 Gold 码中任意两序列间的互相关函数都满足式(1)．换句话说．若定义 $t(n)=2^{[(n+1)/2]}+1$，则 G 族中任意两序列的互相关函数均取 3 值：-1，$-t(n)$ 和 $t(n)-2$．

可见，Gold 码具有较好的相关特性，且满足地址码的要求．

但是，Gold 码中有些不具有平衡性．当码序列中“0”和“1”的数字相差为 1 时，该序列为平衡码序列，否则为非平衡码序列．由于 Gold 码序列的自相关函数为 3 值相关函数，所以其中的码有些是非平衡码．当 n 为奇数时，平衡码和非平衡码出现概率各占 50%；当 n 为偶数时，平衡码为 75%，非平衡码为 25%．

把 Gold 码应用于码分多址技术时，码族中的不平衡性对载波抑制度有很大影响．当码长不是很长时，采用非平衡码对载波抑制度比平衡码下降了一半．从载波抑制角度看，Gold 码可提供使用的地址数不是 2^n+1 个，至多只有其中的 75%(n 为偶数)．

早期总结出的经典方法[4,5]求得的 m 序列优选对，由于当时条件的局限性并非是最优的，即并非是 n 次本原多项式族中互相关最大值的绝对值 $|\rho_{max}|$ 中的最小者．Jack．k．Helmes 应用 R．Cold 论文[4]综述了平衡 Gold 码产生的方法[6]，验证表明，有某些不完善之处．现在，随着科学技术的发展，使用计算机寻找 m 序列优选对与平衡 Gold 码，既准确，又快速方便．

2 m 序列优选对的确定

寻找 m 序列优选对的方法很多．有经典手算法，逐步移位模二加算法(又称硬算法)，分元陪集法及 3 值判别法[7]．这里，给出兼用分元陪集和3 值判别的结合法．

陪集是将$(0,1,2,\cdots,P-1)$，$P=2^n-1$ 这个正整数集进行分类的一种方法．例如，对于 $n=4$ 的情形，共有 5 个陪集：

$C_0:0$　　$C_1:1,2,4,8$

$C_2:3,6,12,9$　　$C_3:5,10$

$C_4:7,14,13,11$

利用陪集的概念，可以证明[8]，两个相同周期的m序列，当τ取同一陪集内各个数值时所得的互相关值相同．就是说，若有两个m序列，它们的周期$P=2^n-1$，且均有k个陪集：C_0，C_1，⋯，C_{k-1}，那么

$$\rho(\tau)=a_i,\ \tau\in C_i,\ i=0,1,\cdots,k-1 \tag{3}$$

于是最多只需求k个互相关函数值．

对于$n=4$，即长为$2^n-1=15$的序列

110101100100011 和 100110101111000

它们的互相关函数只需要计算5个值

$$\rho(\tau)=\begin{cases}-1/15, & \tau\in C_0\\ -1/3, & \tau\in C_1\\ 1/5, & \tau\in C_2\\ 7/15, & \tau\in C_3\\ -1/15, & \tau\in C_4\end{cases}$$

用分元陪集法计算m序列优选对，随着级数n的增大，其优越性更为突出．因为n越大，陪集数k相对于m序列的周期P要小得多，因此，计算量较硬算法要小得多．如果在计算中出现与m序列优选对的3个互相关函数值不同的第4个值时，即判定该序列对不是优选对，便停止运算．

3 程序设计

将分元陪集法和3值判别法结合起来编制FORTRAN程序来确定m序列优选对．程序框图如图3所示．

本程序由主程序和3个子程序组成．

a．陪集(PEIJI)子程序

分元陪集的构造方法是，先将1至$P-1$($P=2^n-1$)中与P互素的所有正整数集合中的1，2，2^2、⋯，2^n-1组成第1个陪集，其次任选一个不曾包含在第一个陪集中的数分别乘上第1陪集的所有项(经模P化简，下同)组成第2陪集，然后再选1个不曾包含在第1和第2陪集中的所有项组成第3陪集，依此继续组成其余各陪集，直至把所有与P互素的正整数用完为止．如果在这个基础上继续把0至$P-1$和整数集合剩下的，与P不互素的数(这次把0包括在内)，也按同样的方法组成其它陪集，便可以得到一些长度缩短的陪集．陪集子程序框图如图4所示．

b．将八进制数化为二进制数子程序(AIG)

将八进制数依次除以$10^{n'-1}$，$10^{n'-2}$，⋯，10^0，用取整的方法求出第n'位，第$n'-1$位，⋯，第1位(其中n'表示八进制数字符号长度)，再将每1位转换成二进制数，将所有的二进制数按n'的次序排在一起即得八进制数的二进制表示．

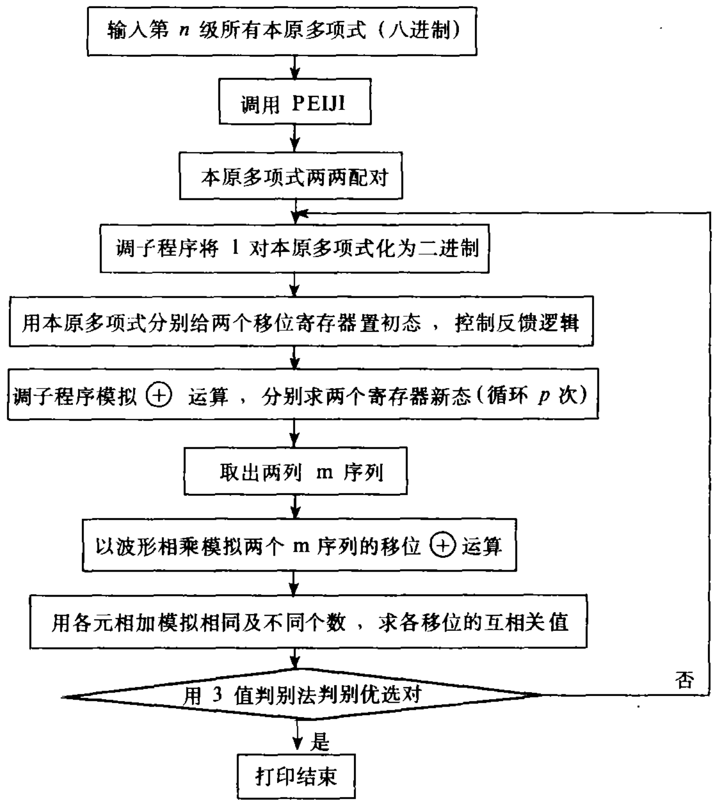

图 3　求 m 序列优选对程序框图

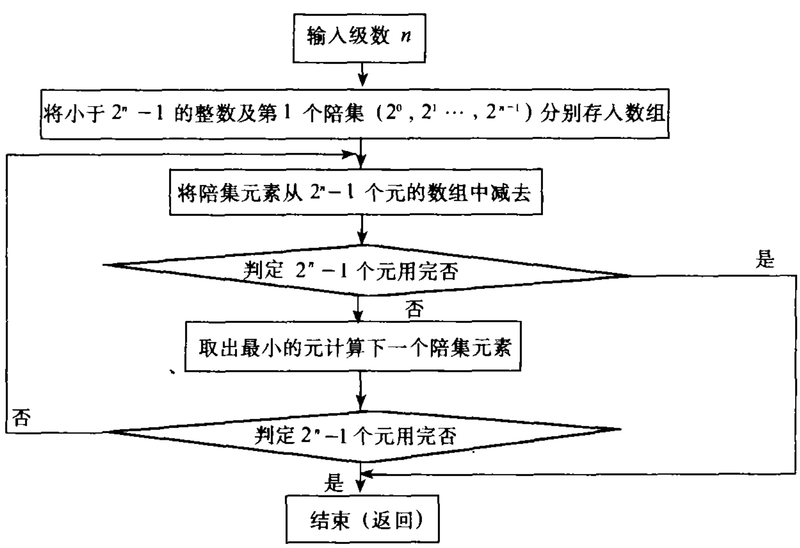

图 4　陪集子程序框图

c．产生m序列子程序(BIG)

将上步得到的二进制数作为移位寄存器的初态及反馈控制逻辑，反馈循环移位(2^n-1)次，即可得到m序列．

d．计算机实现模二加运算

将"0"和"1"组成的序列变换成对应的二进制信号波形，对应关系为

符号序列		波形序列
1	:	−1
0	:	1

在这种对应关系下，符号序列的模二加运算，即计算相关函数值的基本运算对应信号波形的乘法运算．通过这种乘法运算，两序列码元相同者一定得"1"，码元不同者一定得"−1"，再将此积组成的新序列逐位相加就得到相同码元数与不同码元数之差，从而模拟了相关函数值的运算．

最后由3值判别法判定相关函数值是否出现3值以外的数，从而确定是否优选对．

请注意，主程序输入的是本原多项式．已知二元域上，如果n次多项式是本原多项式，那么以此作为联接多项式的n项线性反馈移位寄存器可产生最大周期的序列，即m序列．因此，确定m序列优选对，就意味着确定码发生器的两组寄存器以哪些本原多项式对作为其联结多项式．表1和表2分别给出部分7级和9级的m序列优选对结果．用这些

表1　m序列优选对

211	217，	211	235，	211	247，	211	253，	211	277，
211	357，	211	203，	211	325，	211	323，	211	301，
217	235，	217	247，	217	253，	217	277，	217	357，
217	203，	217	271，	217	325，	217	323，	235	247，
235	277，	235	313，	235	357，	235	361，	235	325，
235	375，	235	301，	247	277，	247	313，	247	357，
247	203，	247	325，	247	323，	247	301，	253	277，
253	203，	253	221，	253	361，	253	271，	253	345，
253	323，	253	367，	277	357，	277	203，	277	271，
277	323，	277	367，	313	357，	313	221，	313	361，
313	345，	313	325，	313	375，	313	367，	313	301，
357	325，	357	375，	357	323，	357	301，	203	221，
203	271，	203	345，	203	323，	203	367，	221	361，
221	271，	221	345，	221	325，	221	375，	221	367，
221	301，	361	271，	361	345，	361	325，	361	375，
361	367，	361	301，	271	345，	271	375，	271	323，
271	367，	345	375，	345	323，	345	367，	345	301，
325	375，	325	301，	375	367，	375	301，	323	367，

注：$n=7$，$|\rho_{max}|=17$

表 2 m 序列优选对

1 021	1 033、	1 021	1 055，	1 021	1 167，	1 021	1 243、	1 021	1 257，
1 021	1 317、	1 021	1 333、	1 021	1 533，	1 021	1 617、	1 021	1 131，
1 021	1 461、	1 021	1 605，	1 021	1 245，	1 021	1 665、	1 021	1 063，
1 033	1 157、	1 033	1 207、	1 033	1 225，	1 033	1 257、	1 033	1 577，
1 033	1 371、	1 033	1 365、	1 033	1 715，	1 033	1 707，	1 033	1 151，
1 033	1 751，	1 055	1 175、	1 055	1 275，	1 055	1 333，	1 055	1 577，
1 055	1 131，	1 055	1 137，	1 055	1 731，	1 055	1 245，	1 055	1 425、
1 055	1 665，	1 055	1 443，	1 055	1 563，	1 055	1 713、	1 063	1 157，
1 063	1 167，	1 063	1 517，	1 063	1 533，	1 063	1 617，	1 063	1 041、
1 063	1 605、	1 063	1 665，	1 063	1 443，	1 063	1 563，	1 063	1 773，
1 157	1 175，	1 157	1 257，	1 157	1 317，	1 157	1 517，	1 157	1 617、
1 157	1 131、	1 157	1 137、	1 157	1 321，	1 157	1 671、	1 157	1 245，
1 157	1 425，	1 157	1 365，	1 157	1 555，	1 167	1 207，	1 167	1 243，

注：$n=9$，$|\rho_{max}|=33$

本原多项式对作为码发生器的联结多项式产生的一定是 m 序列优选对．而 m 序列优选对模二加即产生 Gold 序列．

上述的几个子程序在后面的平衡 Gold 码程序中也可直接调用．

4 平衡Gold 序列的实现

m 序列优选对的相对相移量不同，得到的 Gold 序列也不同，其中的非平衡码不能做地址使用．m 序列优选对的相对相移量与码发生器的两组寄存器的初始状态有关．因此，我们希望知道寄存器在哪些初态下能产生平衡 Gold 序列．图 5 给出平衡 Gold 序列程序框图．从而，可求得任意级数的码发生器，在第 1 组寄存器初态为全“1”时，第 2 组以哪些状态为初态，便可产生平衡 Gold 序列．

本程序包括主程序和 4 个子程序

a．主程序：读入寄存器级数和m序列优选对，并调用各子程序完成平衡 Gold 码发生器第 2 组寄存器初始状态的选择．

b．子程序AIG：前已叙及．

c．子程序BIG：前已叙及．

d．子程序DIG：判断Gold 是否为平衡序列．如是，则将第 2 组寄存器状态打印输出．

e．子程序CIG：实现两寄存器组初始状态相对相移的改变，即将第2 组寄存器的状态反

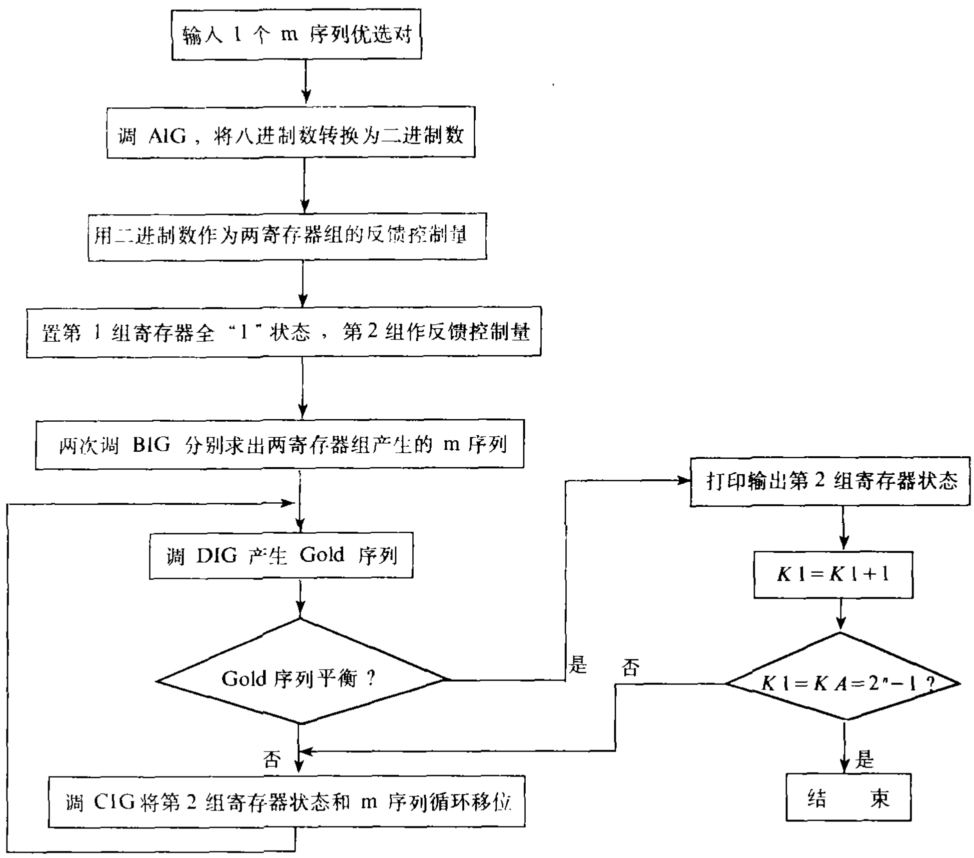

图 5　平衡 Gold 序列程序框图

馈跳变到下一状态．由m序列的产生可知，码发生器初始状态跳变1次，它产生的m序列也相应地循环移位1次．因此，本子程序模拟实现了m序列的循环移位．

表3～5分别给出，在输出为平衡Gold码时n=7，9，10第2组寄存器初态的部分计算结果．

表 3　Gold 码发生器可预置状态

0000101	0001010	0010101
1011000	1000100	0001001
0010011	1001111	0111100
1111001	1100101	1001001
0100101	0010110	1011010

注：n=7、m 优选对：211 217

表 4　Gold 码发生器可预置状态

110000001	000000101	000001010
000101001	010011101	100111010
110101001	101010011	010011001
100110010	001100101	010111101

注：n=9、m 优选对：1021　1033

表 5　Gold 码发生器可预置状态

1011000010	0110000100	1000010011
0000100110	0001001101	0100110110
1001101101	0011011010	0110110101
1011010111	0110101110	1101011100

注：n=10、m 优选对：2011　2415

5 结束语

本文主要叙述了用计算机确定m序列优选对和平衡Gold序列条件下的第2组寄存器的初始状态．其理论依据是用分元陪集和3值判别相结合的方法求m序列优选对，再由所得的优选对选择平衡Gold序列状态下的第2组寄存器的初态．这种方法较之以往文献公布的方法计算量要小，运算速度提高，且计算结果准确，是目前较好的方法．

参考文献

1 Dixon R C . Spread spectrum systems . 2nd ed . New York : John Wiley & Sons , 1984 .

2 李振玉，卢玉民．扩频选址通信．北京：国防工业出版社，1988．

3 肖国镇等．伪噪声序列及其应用．北京：国防工业出版社，1985．

4 Gold R . Maxmal recursive sequences with 3 - valued recursiv cross - correlation functions . IEEE Trans . Information Theory , 1968 .

5 万哲先．代数和编码．北京：科学出版社，1985．

6 Holmes J K . Coherent spread spectram systems . New York : John Wiley & Sons , 1982 .

7 辛光明，潘梦菡．m序列优选对的计算实现及平衡戈尔德码组．成都电讯工程学院学报，1985，增刊(二)：34～44

8 钟义信．伪噪声编码通信．北京：人民邮电出版社，1979．

Optimum m - Sequence Pairs and Blanced Gold Group

Xin Xiaoming　Chen Qiong

(Department of Electronic Engineering)

Abstract Gold code is now widely used in spread spectrum addressing comunications . However , the m - sequence pairs adopted are usually not optimum , or even unbalanced . This paper provides an algorithm for the realization of optimum m-sequence pairs and balanced Gold codes with the help of a computer . The amount of computation is much redueed and the high speed in calculation is supported by a high degree of accuracy . Results of calculation show that it is an algorithm worthy of recommendation .

Key words M - sequence / m - sequence , Gold code , balanced Gold code .