信息论与编码

马啸
maxiao@mail.sysu.edu.cn

计算机学院
中山大学

2021 年春季学期

有限域的特征与阶

设 $\mathbb{F}_q$ 是一个有限域。从定义，我们知必有一个元素“1”，即乘法的单位元素。考虑下面的序列

$$1,\ 1+1,\ 1+1+1,\ \cdots,\ \underbrace{1+1\cdots+1}_{n},\ \cdots$$

由于 $\mathbb{F}_q$ 是有限的，所以必有 $i<j$ 满足

$$\underbrace{1+1\cdots+1}_{i}=\underbrace{1+1\cdots+1}_{j}$$

在此条件下，有 $\underbrace{1+1\cdots+1}_{j-i}=0$。我们记最小的正整数 p，使得

$$\underbrace{1+1\cdots+1}_{p}=0$$

可以证明 p 是素数。

有限域的特征与阶

事实上，若 p 不是素数，则 $p = ab$，$a > 1$，$b > 1$，我们有 $\underbrace{(1+1\cdots+1)}_{a}\underbrace{(1+1\cdots+1)}_{b} = 0$，则必有 $a \cdot 1 = 0$ 或 $b \cdot 1 = 0$，与 p 最小矛盾。

我们记

$$\mathbb{F}_p = \{0, 1, 1+1, \cdots, \underbrace{1+1+\cdots+1}_{p-1}\} \triangleq \{0, 1, 2, \cdots, p-1\}.$$

有限域的特征与阶

我们可以验证 $\mathbb{F}_p$ 是域，与模 p 运算下定义的域是同构的。p 称之为有限域 $\mathbb{F}_q$ 的特征。显然，对于任意的 $\alpha \in \mathbb{F}_q$，有 $p \cdot \alpha = 0$。

下面我们说明，$\mathbb{F}_q$ 可以看作 $\mathbb{F}_q$ 上的线性空间：$\forall \alpha, \beta, \gamma \in \mathbb{F}_q$, $\forall r, s \in \mathbb{F}_p$.

1. $\alpha + \beta = \beta + \alpha$
2. $(\alpha + \beta) + \gamma = \alpha + (\beta + \gamma)$
3. $\alpha + 0 = 0 + \alpha = \alpha$
4. $\exists x$, 使得$\alpha + x = 0$
5. $1 \cdot \alpha = \alpha$
6. $(r \cdot s)\alpha = r \cdot (s\alpha)$
7. $(r + s)\alpha = r\alpha + s\alpha$
8. $r(\alpha + \beta) = r\alpha + r\beta$

有限域的特征与阶

既然 $\mathbb{F}_q$ 是 $\mathbb{F}_p$ 上的线性空间，且 $\mathbb{F}_q$ 是有限的，所以其维数有限，记为 m。设 α_1，α_2，$\cdots$，α_m 是一组基，则 $\forall \alpha \in F_q$，均可表示为

$$\alpha = \sum k_i \alpha_i, \ k_i \in F_p$$

因此 $\mathbb{F}_q$ 的阶是 p^m。

在特征为 p 的有限域中，有许多性质看起来与实数域不同，比如 $(x+y)^p = x^p + y^p$。特别地，在 $\mathrm{GF}(2^m)$ 中，$(x+y)^2 = x^2 + y^2$。

$GF(p^m)$的构造

一般$GF(p^m)$ 是通过$GF(p)$ 上的多项式来构造的。考虑系数取自$GF(p)$ 上多项式

$$f(x) = \alpha_0 + \alpha_1 x + \cdots + \alpha_n x^n$$

其中$\alpha_i \in GF(p), i = 0, \cdots, n$。若$\alpha_n \neq 0$, 称$f(x)$ 为n 次多项式且记做$\partial^\circ f(x) = n, \alpha_n$ 称为首项系数。若$\alpha_n = 1$,称为首一多项式。常数(不为0) 可以看做是 0 次多项式。0 多项式的次数认为是 $-\infty$。

$GF(p^m)$的构造

多项式加法和乘法定义如下。若

$$f(x)=\sum_{i=0}^{n}\alpha_i x^i, g(x)=\sum_{i=0}^{m}\beta_i x^i$$

加法定义为

$$f(x)+g(x)=\sum_{i=0}^{\max(m,n)}(\alpha_i+\beta_i)\,x^i$$

如果$m>n$, 则$i>n$ 时认为$\alpha_i=0$ 。若$n>m$, 则$i>m$ 时认为$\beta_i=0$ 。
乘法定义为

$$f(x)g(x)=\sum_{i=0}^{m+n}c_i x^i$$

其中

$$c_i=\sum_{j=0}^{i}\alpha_j\beta_{i-j}$$

$GF(p^m)$的构造

与整数类似,多项式除法采用带余除法

$$a(x) = q(x)b(x) + r(x), \partial^\circ r(x) < \partial^\circ b(x)$$

记做$a(x) \bmod b(x) = r(x)$ 。多项式模运算也有下列性质

$$\begin{aligned} a_1(x) \bmod b(x) + a_2(x) \bmod b(x) &= [a_1(x) + a_2(x)] \bmod b(x) \\ [a_1(x) \bmod b(x)]\,[a_2(x) \bmod b(x)] &= [a_1(x)a_2(x)] \bmod b(x) \end{aligned}$$

如果有$f(x) = q(x)(x - \alpha)$, 则$f(\alpha) = 0$,称α 为$f(x)$ 的根。α 为$f(x)$ 的根的充要条件是$(x - \alpha)$ 是$f(x)$ 的因式,记为$(x - \alpha) \mid f(x)$ 。
如果$f(x)$ 在$GF(p)$ 上仅能被不为0 的常数,或者自身的常数倍整除,不能被其他多项式除尽, $f(x)$ 称为既约多项式,其定义与整数中素数的概念类似。

$GF(p^m)$的构造

任意两个多项式$f(x)$ 和$g(x)$, 以$(f(x), g(x))$ 表示它们的最大公因式(首一多项式),它可由$f(x)$ 和$g(x)$ 的线性组合表示

$$(f(x), g(x)) = a(x)f(x) + b(x)g(x)$$

其中, $f(x)$ 和$g(x)$ 的最小公倍式(首一多项式) 记做$[f(x), g(x)]$,有

$$f(x)g(x) = (f(x), g(x))[f(x), g(x)]$$

$GF(p^m)$的构造

如果$(f(x), g(x)) = 1$,称$f(x)$ 和$g(x)$ 互素。此时

$$1 = a(x)f(x) + b(x)g(x)$$

当以$g(x)$ 为模时，有

$$1 \bmod g(x) = [a(x) \bmod g(x)][f(x) \bmod g(x)]$$

可以根据域的定义验证得出,若$g(x)$ 为既约多项式,以$g(x)$ 为模时,与$g(x)$ 互素的多项式构成域。若$g(x)$ 为m 次多项式, 以$g(x)$ 为模的多项式剩余类(次数小于m 的所有多项式集合)构成一个域$GF(p^m)$,因为这样的多项式个数为p^m 个。

例子：$GF(2^4)$的构造

取$GF(2)$ 上的4 次既约多项式$p(x)=1+x^3+x^4$ 。设$\alpha \in GF\left(2^4\right)$ 是$p(x)$ 的一个根，即有$p(\alpha)=1+\alpha^3+\alpha^4=0$ 。$GF\left(2^4\right)$ 上的元素将有多项式和幂两种表示方式。

根据前面的描述，$GF(2)$ 上次数小于4 的所有多项式构成$GF\left(2^4\right)$ 。因为我们将用多项式的$a_0+a_1x+a_2x^2+a_3x^3$ 系数表示$GF\left(2^4\right)$ 的元素。这称为多项式表示。

以$p(x)$ 的根α 的幂次$\alpha^i\,(i=0,1,\cdots,2^m-2)$ 表示$GF\left(2^4\right)$ 的非零元素,称为元素的幂表示。以幂表示除以$p(x)$ 所得的余式即为相应的多项式表示。由$p(x)=1+x^3+x^4$ 构造的$GF\left(2^4\right)$ 元素的幂表示和多项式表示如表所示。

例子：$GF(2^4)$的构造

系数$a_0a_1a_2a_3$	多项式	幂表示
(0000)	0	0
(1000)	1	$1=\alpha^0$
(0100)	α	α
(0010)	α^2	α^2
(0001)	α^3	α^3
(1100)	$1+\alpha^3$	α^4
(1101)	$1+\alpha+\alpha^3$	α^5
(1111)	$1+\alpha+\alpha^2+\alpha^3$	α^6
(1110)	$1+\alpha+\alpha^2$	α^7
(0111)	$\alpha+\alpha^2+\alpha^3$	α^8
(1010)	$1+\alpha^2$	α^9
(0101)	$\alpha+\alpha^3$	α^{10}
(1011)	$1+\alpha^2+\alpha^3$	α^{11}
(1001)	$1+\alpha$	α^{12}
(0110)	$\alpha+\alpha^2$	α^{13}
(0011)	$\alpha^2+\alpha^3$	α^{14}

例子：$GF(2^4)$的构造

GF(2^m) 中元素的加法为普通的多项式加法,乘法为模$p(x)$ 的多项式乘法。系数之间的运算为表所示的$GF(2)$ 上的运算。如$GF\left(2^4\right)$ 中

$$\alpha^5+\alpha^7=1+\alpha+\alpha^3+1+\alpha+\alpha^2=\alpha^2+\alpha^3=\alpha^{14}$$
$$\alpha^5\cdot\alpha^7=\alpha^{12}=\left(1+\alpha+\alpha^3\right)\left(1+\alpha+\alpha^2\right)\bmod 1+\alpha^3+\alpha^4=1+\alpha$$

利用幂表示，$GF\left(2^m\right)$ 的乘法运算可按照下面方法非常方便地计算

$$\alpha^i\cdot\alpha^j=\alpha^{(i+j)\bmod 2^m-1}$$

循环码的数学描述

Definition 1

一个(n, k) 线性码C, 若对任意$c = (c_0, c_1, \cdots, c_{n-1}) \in C$, 恒有$c' = (c_{n-1}, c_0, \cdots, \mathbf{c}_{n-2}) \in C$, 称$C$ 为循环码。

循环码的码字可以用向量表示之外,还可以用x 的多项式表示为

$$c(x) = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1}$$

称$c(x)$ 为码字多项式。若以x 乘以$c(x)$,并用多项式$x^n - 1$ 去除可以得到

$$\begin{aligned} xc(x) &= \left(c_0 x + c_1 x^2 + \cdots + c_{n-1} x^n\right) \bmod (x^n - 1) \\ &= c_{n-1} + c_0 x + \cdots + c_{n-2} x^{n-1} \end{aligned}$$

这样,循环码的循环移位可由模$x^n - 1$ 下的码字多项式$c(x)$ 乘以x 运算给出,循环码的研究可以利用模$x^n - 1$ 的多项式代数R_n 进行。R_n 是$GF(q)$ 上低于n 次的所有q^n 个多项式的集合。

循环码的数学描述

Theorem 2

若$c(x)$ 为一个循环码多项式, $b(x) \in R_n$, 则$b(x)c(x) \bmod x^n - 1$ 也是一个码多项式。

证明： 令$b(x) = b_0 + b_1 x + \cdots + b_{n-1}x^{n-1}$, 由于循环码的定义可知, $xc(x), x^2c(x), \cdots, x^{n-1}c(x)$ 都是码多项式。又因为循环码是线性码,这些码字的线性组合还是循环码的一个码字, 因此$b_0c(x) + b_1xc(x) + \cdots + b_{n-1}x^{n-1}c(x)$ 是一个码多项式，证毕。

定理说明码多项式的倍式仍是一个码多项式。

循环码的数学描述

在一个循环码当中,存在一个次数最低的非零多项式$g(x)$,该多项式的最高次项系数为1，称为首一多项式。对$g(x)$ 有下述定理。

Theorem 3

循环码中的所有码多项式$c(x)$ 都是$g(x)$ 的倍式, $g(x)$ 称为循环码的生成多项式而且是唯一的。

证明： 设$g(x)$ 是循环码C 中次数最低的首一多项式,对任意的$c(x) \in C$, 有$c(x) = q(x)\ g(x) + r(x)$, $r(x)$ 是$c(x)$ 除$g(x)$ 的余式, 所以, $r(x)$ 次数小于$g(x)$ 的次数。由于$q(x)g(x) \in C$,有$r(x) = c(x) - q(x)g(x) \in C$, 与$g(x)$ 是循环码C 中次数最低的首一多项式假设矛盾,所以$r(x) = 0$, 即所有码多项式都是$g(x)$ 倍式。
设$h(x)$ 和$g(x)$ 都是C 中次数最低的首一多项式,则由$h(x) - g(x) \in C$ 知,C 中有次数更低的多项式存在,与假定矛盾。因此C 中次数最低的首一多项式是唯一的，证毕。

循环码的数学描述

Theorem 4

若C 是R_n 中的循环码,则C 的最低次首一生成多项式$g(x)$ 是x^n-1 的因式。

证明： 令$g(x)$ 是C 的最低次首一多项式,则有$x^n-1=q(x)g(x)+r(x)$, 其中$r(x)$ 的次数低于$g(x)$ 的次数。而$r(x)=q(x)g(x) \bmod x^n-1$, 所以$r(x)$ 在C 中。由于$g(x)$ 是C 中次数最低的首一多项式，所以必有$r(x)=0$, 因此$g(x)$ 是x^n-1 的因式。

循环码的数学描述

由上述几个定理知道,构造循环码在于从分解x^n-1 的分解中取出一个因式$g(x)$, 以它作为生成多项式就可构成一个循环码。x^n-1 的分解可以查表[Peterson(1961)]。
给定一个$n-k$ 次首一多项式$g(x)$, ,则$g(x), xg(x), x^2g(x), \cdots, x^{k-1}g(x)$ 的系数矢量是线性独立的,由它们可以构成码的生成矩阵

$$\mathbf{G}=\begin{bmatrix} g_0 & g_1 & g_2 & \cdots & g_{n-k} & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & \cdots & g_{n-k-1} & g_{n-k} & \cdots & 0 \\ \vdots & & \ddots & & & & \ddots & \vdots \\ 0 & 0 & \cdots & g_0 & & \cdots & & g_{n-k} \end{bmatrix}$$

循环码的数学描述

如果将待编码的消息序列$\mathbf{u} = (u_0, u_1, \cdots, u_{k-1})$ 表示为多项式

$$u(x) = u_0 + u_1 x + \cdots + u_{k-1} x^{k-1}$$

则以x^{n-k} 乘以$u(x)$,再以$g(x)$ 除,得

$$x^{n-k} u(x) = q(x) g(x) + r(x)$$

其中$q(x)$ 是商式。$r(x) = r_0 + r_1 x + \cdots + r_{n-k-1} x^{n-k-1}$ 是余式。次数低于$g(x)$ 的次数。重新排列上式,得

$$c(x) = x^{n-k} u(x) - r(x) = q(x) g(x) \in C$$

其中$c(x)$ 的系数为$(-r_0, -r_1, \cdots, -r_{n-k-1}, u_0, u_1, \cdots, u_{k-1})$ 正是系统码形式的码字。由此得出系统循环码的编码步骤如下:

1. 以x^{n-k} 乘以$u(x)$ 。
2. 以$g(x)$ 除$x^{n-k} u(x)$ 得余式$r(x)$ 。
3. 组合$r(x)$ 和$x^{n-k} u(x)$ 得到码字$[-r(x), u(x)]$。

循环码的数学描述

Example 5

$q=2$ 时选$g(x)=1+x+x^3$，它除尽$1-x^7$ 。若$u(x)=1+x^3$，则由

$$x^3\left(1+x^3\right)=x^3+x^6=\left(x+x^2\right) \bmod g(x)$$

得出码多项式$c(x)=x+x^2+x^3+x^6$，即码字$c=(0111001)$。

循环码的数学描述

设 $x^n - 1 = g(x)h(x)$, $g(x) = \sum_{i=0}^{n-k} g_i x^i, h(x) = \sum_{i=0}^{k} h_i x^i$. 由 $g(x)$ 生成的循环码的生成矩阵是

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & \cdots & g_{n-k} & & \cdots & 0 \\ 0 & g_0 & \cdots & g_{n-k-1} & g_{n-k} & \cdots & 0 \\ & \ddots & \ddots & & & \ddots & \vdots \\ 0 & \cdots & 0 & g_0 & & \cdots & g_{n-k} \end{bmatrix}$$

循环码的数学描述

其校验矩阵可以按如下方式推导

Theorem 6

由 $x^n - 1 = g(x)h(x)$, 我们可以得到：

$$
\begin{array}{l}
g_0 h_0 = 1 \\
\sum_{l=0}^{n-k} g_l h_{i-l} = 0, for 1 \leqslant i \leqslant n-1 \\
g_{n-k} h_k = 1
\end{array}
$$

$$
\mathbf{H} = \begin{bmatrix}
h_k & h_{k-1} & \cdots & h_0 & & \cdots & 0 \\
0 & h_k & \cdots & h_1 & h_0 & \cdots & 0 \\
 & \ddots & \ddots & & & \ddots & \vdots \\
0 & \cdots & 0 & h_k & & \cdots & h_0
\end{bmatrix}
$$

G 的第一行与 H 的各行的“内积”依次是 $x^k, x^{k+1}, \cdots, x^{n-1}$ 的系数。

循环码的数学描述

Example 7

由$x^7-1=(1+x)\left(1+x+x^3\right)\left(1+x^2+x^3\right)$, 例7 选$g(x)=1+x+x^3$,
则$h(x)=(1+x)\left(1+x^2+x^3\right)=1+x+x^2+x^4$, 即

$$h^*(x)=x^4\left(1+x^{-1}+x^{-2}+x^{-4}\right)=1+x^2+x^3+x^4$$

由此得

$$\mathbf{H}=\begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

由式$u(x)=u_0+u_1x+\cdots+u_{k-1}x^{k-1}$, 当给定$\mathbf{u}=(u_0,u_1,\cdots,u_{k-1})$ 时,按多项式可计算

$$x^{n-k}u(x)=u_0x^{n-k}+u_1x^{n-k+1}+\cdots+u_{k-1}x^{n-1}$$

其系数矢量和$\mathbf{H}$ 矩阵相应的最后一行的内积为0 可算出码的相应位c_{n-k-1}, 即

$$c_{n-k-1}h_k+u_0h_{k-1}+\cdots+u_{k-1}h_0=0$$

因为$h^*(x)$ 是首一多项式,所以$h_k=1$ 。因此

$$c_{n-k-1}=u_0h_{k-1}+\cdots+u_{k-1}h_{00}$$

然后将已知的$k+1$ 个高次项系数组成矢量与$\mathbf{H}$ 矩阵的倒数第二行进行内积运算,令其结果为0 可以算出c_{n-k-2}。依此类推就可以得到系统码的码字。

常见通信标准中的CRC

- 以太网–CRC32 **[IEEE 802.3 from Wikipedia]**:

$$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x+1$$

- 5G NR **[3GPP TS 38.212 V15.8.0 (2019-12)]**:

5.1 CRC calculation

Denote the input bits to the CRC computation by $a_0, a_1, a_2, a_3, \ldots, a_{A-1}$, and the parity bits by $p_0, p_1, p_2, p_3, \ldots, p_{L-1}$, where A is the size of the input sequence and L is the number of parity bits. The parity bits are generated by one of the following cyclic generator polynomials:

- $g_{\mathrm{CRC24A}}(D)=[D^{24}+D^{23}+D^{18}+D^{17}+D^{14}+D^{11}+D^{10}+D^{7}+D^{6}+D^{5}+D^{4}+D^{3}+D+1]$ for a CRC length $L=24$;
- $g_{\mathrm{CRC24B}}(D)=[D^{24}+D^{23}+D^{6}+D^{5}+D+1]$ for a CRC length $L=24$;
- $g_{\mathrm{CRC24C}}(D)=[D^{24}+D^{23}+D^{21}+D^{20}+D^{17}+D^{15}+D^{13}+D^{12}+D^{8}+D^{4}+D^{2}+D+1]$ for a CRC length $L=24$;
- $g_{\mathrm{CRC16}}(D)=[D^{16}+D^{12}+D^{5}+1]$ for a CRC length $L=16$;
- $g_{\mathrm{CRC11}}(D)=[D^{11}+D^{10}+D^{9}+D^{5}+1]$ for a CRC length $L=11$;
- $g_{\mathrm{CRC6}}(D)=[D^{6}+D^{5}+1]$ for a CRC length $L=6$.

常见通信标准中的CRC

- LTE(4G+) **[3GPP TS 36.212 V10.9.0 (2015-09)]:**
  - 5.1.1 CRC calculation

Denote the input bits to the CRC computation by $a_0, a_1, a_2, a_3, ..., a_{A-1}$, and the parity bits by $p_0, p_1, p_2, p_3, ..., p_{L-1}$. A is the size of the input seque and L is the number of parity bits. The parity bits are generated by one of the following cyclic generator polynomials:

- $g_{\text{CRC24A}}(D) = [D^{24} + D^{23} + D^{18} + D^{17} + D^{14} + D^{11} + D^{10} + D^7 + D^6 + D^5 + D^4 + D^3 + D + 1]$ and;
- $g_{\text{CRC24B}}(D) = [D^{24} + D^{23} + D^6 + D^5 + D + 1]$ for a CRC length $L = 24$ and;
- $g_{\text{CRC16}}(D) = [D^{16} + D^{12} + D^5 + 1]$ for a CRC length $L = 16$.
- $g_{\text{CRC8}}(D) = [D^8 + D^7 + D^4 + D^3 + D + 1]$ for a CRC length of $L = 8$.

循环码的译码

循环码译码步骤和线性码一样,也是先计算接收矢量的伴随式, 然后根据它判断是否有错，当有错时进而判断错误图样并进行纠正。由于循环码的循环结构使得它的译码可以较线性码简单而易于实现。

设接收矢量$v = (v_0, v_1, \cdots, v_{n-1}) \in V_n(q)$, 相应多项式表示为$v(x) = v_0 + v_1 x + \cdots + v_{n-1}x^{n-1}$, 以$g(x)$ 除得

$$v(x) = q(x)g(x) + s(x) = s(x) \bmod g(x)$$

称低于$n-k$ 次的余式$s(x) = s_0 + s_1 x + \cdots + s_{n-k-1}x^{n-k-1}$ 为接收多项式$v(x)$ 的伴随多项式。若$s(x) = 0 \bmod g(x)$,就认为无错(可能含有不可检错误图样)。若$s(x) \neq 0 \bmod g(x)$, 则$v(x)$ 中必有错误存在,如果将错误图样表示成多项式$e(x) = e_0 + e_1 x + \cdots + e_{n-1}x^{n-1}$, 则接收多项式$v(x) = c(x) + e(x)$,相应的伴随多项式为$s(x) = v(x) = c(x) + e(x) = e(x) \bmod g(x)$,即它只与信道错误图样$e(x)$ 有关。$s(x)$ 的计算电路类似于以$g(x)$ 为模的除法电路。

循环码的译码

Theorem 8

若$s(x)$ 是$v(x)$ 的伴随多项式,则$s^{(1)}(x) = xs(x) \bmod g(x)$ 是$v^{(1)}(x) = xv(x) \bmod (x^n - 1)$ 的伴随多项式。

证明： 以x 乘$v(x) = q(x)g(x) + s(x)$ 的两边,并以$g(x)$ 为模,则左边为$s^{(1)}(x)$, 右边即为

$$xs(x) \bmod g(x).$$

循环码的译码

由于伴随式循环移位在模$g(x)$ 下的结果正好和码字的循环移位相对应,这使得对循环码字任何一位的译码都是循环等价的,因而循环码可以逐位进行译码,如下述。

1. 由$v(x)$ 计算伴随式$s(x)$ 。根据$s(x)$ 识别$v_{n-1} \neq 0$ 的可纠正错误图样,若断定这类图样存在,就对v_{n-1} 位的错误进行纠正,得到$v'(x) = v(x) - e_{n-1}x^{n-1}$ 。对$s(x)$ 进行修正,即将$e_{n-1}x^{n-1} \bmod g(x)$ 的值加到$s(x)$ 上,得到与$v(x)$ 相应的$s'(x)$,转入(2) 。若识别结果为$e_{n-1} = 0$, 就直接转入(2)。
2. 将接收数据寄存器和伴随式电路各循环移位一次得到$v^{(1)}(x)$ 和$s^{(1)}(x)$ 。然后对v_{n-2} 得类似于(1) 的处理,依此类推。
3. 循环移位n 次后,若$s^{(n)}(x) = 0 \bmod g(x)$,就将接收缓存器的存数作为$\hat{c}(x)$ 送出。若$s^{(n)}(x) \neq 0 \bmod g(x)$, 就说明译得的结果中必有不可纠正的错误图样。

循环码的译码

Example 9

对以$GF(2)$ 上的多项式$g(x)=1+x+x^3$ 生成的二元$(7,4)$ 码,令接收矢量为$v(x)=v_0+v_1x+\cdots+v_6x^6$,其译码过程可由表说明。
由不难看出,只有当$e(x)$ 的重量为1, 且在首位, 即x^6 上有错时,才会出现$(s_0,s_1,,\ s_2)=(101)$,所以,在循环移位过程中,错误检测器应检验101 的出现,此时相应矢量的首位有错,即可进行纠正。此$(7,4)$ 码的译码电路如图所示。

表: 梅吉特译码器工作过程

错误图样 $e(x)$	伴随式 $s(x)$	伴随式矢量 $s_0s_1s_2$	移位次数 i	i 次移位后的伴随式矢量$s_0s_1s_2$
x^6	$1+x^2$	101	0	101
x^5	$1+x+x^2$	111	1	101
x^4	$x+x^2$	011	2	101
x^3	$1+x$	110	3	101
x^2	x^2	001	4	101
x^1	x	010	5	101
x^0	1	100	6	101

作业

Exercise 1.
找一个 5 次不可约多项式，构造GF(32)，给出元素向量表示与幂表示对应表。

Exercise 2.
列出所有码长是7的二元循环码。

谢谢！