信息与编码理论

——用于通信的数学结构

〔美〕R.J.麦克埃里斯著
刘立柱　译
米鹤颐审校

中国人民解放军工程技术学院情报室

1985

目　　录

第二部分 编码理论

引　论

一九四八年，仙农[1]*在他的经典论文"通信的数学理论"的引言中写道：

"通信的基本问题是在一地准确地或近似地恢复在另一地所选用的消息。"

为了解决这个问题，接下去他在这篇论文中创立了一个崭新的应用数学的分支——今天称为信息理论和编码理论。本书是想在这个理论经历了30年的历史发展之后，对它的主要成果作一介绍。

在这一章里，我们借助于一对特定的数学模型——**二元对称信源和二元对称信道**，来阐明信息理论的中心思想。

二元对称信源（简称信源）是一个客体，它以每单位时间R个符号的速率发出"0"和"1"符号。我们称这些符号为bits，它是binary digtis的缩写。由信源发出的这些比特（bits）是随机的，而且发出"0"和"1"的可能性是相等的。我们设想信源的速率R是连续变量，即R可取任意非负数。

二元对称信道（简记为BSC[2]）是一个客体，它每单位时间可传输一个比特。但是，信道并不是完全可靠的：存在一个固定的概率p，$0 \leqslant p \leqslant \frac{1}{2}$（称为**原误码率**[3]或**原比特错误**

---

*注：上标给出的注释列在每一章的末尾。

概率），即输出比特不等同于输入比特。

我们设想有一个发送者和一个接收者。发送者必然力求把信源的输出尽可能准确地传送给接收者，而二者之间使用的通信线路只是BSC。（但是，在信源接通前，接收者与发送者之间予先知道对方使用的数据处理策略）。我们假定发送者和接收者都拥有足够的计算能力和储存容量，都得到政府的大量资金和其他物资的资助。

现在要问，对于给定的信源速率R，发送者和接收者在BSC上进行通信有多高的准确性？对这个问题，我们终久要给出一个准确的一般答复，但是，我们首先考虑一些特殊情形。

设R = 1/3，意即，信道上传送的比特速率是信源产生比特速率的三倍，因此信源输出在传输之前可按每比特重复三次而编码。例如，信源输出的前5个比特为10100，则编码流为111000111000000。接收者收到的将是这个编码流，而由于信道“噪声”的干扰，可能使与每一信源比特相对应的三个码流比特不尽相同。若信道干扰使第2、第5、第6、第12和第13个传输比特发生错误，则接收者收到的是101011111001100。不难想到，在这种情况下，接收者译出一个给定的信源比特的最佳译码策略是对它的三次重复码采取择多判决。在上述例子中，应将接收到的消息译为11100，这时对第二个比特作出了错误判决。一般地说，一个信源比特三次重复中的两个或三个受到信道干扰的话，它将被错误地接收。于是，若用Pe表示误码率，则

$$Pe = P\{2个传输错误\} + P\{3个传输错误\}$$

$$= 3p^2(1-p) + p^3 = 3p^2 - 2p^3 \qquad (0.1)$$

因为原误码率$p \leqslant \frac{1}{2}$，所以Pe小于p；这个简单的编码方法已提高了信道的可靠性，而当p很小时，可靠性的提高是显著的。

不难看出，用多次重复每个比特的方法可以获得更高的可靠性。因此，若对某一整数n，$R=1/(2n+1)$，在传输之前可以重复每一信源比特（$2n+1$）次（见习题0.2），应用上述的择多判决译码，由此不难得到所产生的误码率$P_e^{(2n+1)}$的公式：

$$P_e^{(2n+1)} = \sum_{k=n+1}^{2n+1} P\{2n+1\text{个传输比特中}k\text{个传输错误}\}$$

$$= \sum_{k=n+1}^{2n+1} \binom{2n+1}{k} p^k (1-p)^{2n+1-k}$$

$$= \binom{2n+1}{n+1} p^{n+1} + p\text{的更高次幂项}。 \quad (0.2)$$

若$n>1$，当$p \to 0$时，$P_e^{(2n+1)}$比$n=1$时更迅速地趋于0。[4]因而，在这个意义上，较长的重复码比较短的更有效。但是，我们希望得出一个更强的论断：对一个确定的BSC，它有确定的原误码率$p<\frac{1}{2}$，则当$n \to \infty$时，$P_e^{(2n+1)} \to 0$，也就是说，通过这种重复码，信道可以得到所需要的可靠性。考察$P_e^{(2n+1)}$的表示式（0.2），我们说，可以做到这一点，但并不容易。因此，我们将用另外的方法。**弱大数定律***指出，若在信道上传输N个比特，则对任意的$\varepsilon>0$，

---

* 在附录A中讨论。

$$\lim_{N\to\infty} P\left\{ \left| \frac{\text{传输错误的个数}}{N} - p \right| > \varepsilon \right\} = 0。 \quad (0.3)$$

换言之，当N大时，被错误接收的比特的比例与p不会有很大的差异。于是我们对$P_e^{(2n+1)}$可作如下估算：

$$\begin{aligned} P_e^{(2n+1)} &= P\left\{ \text{错误接收比特的比例} \geqslant \frac{n+1}{2n+1} \right. \\ &\quad \left. = \frac{1}{2} + \frac{1}{4n+2} \right\} \\ &\leqslant P\{\text{比例} > \tfrac{1}{2}\} \\ &\leqslant P\{\,|\text{比例} - p| > \tfrac{1}{2} - p\} \end{aligned}$$

而由（0.3）式，当$n\to\infty$时，$P_e^{(2n+1)}\to 0$。于是，我们得到了这样的结论：若R很小，即使信道自身是强干扰的，也可能使总的错误概率很小。当然，这并不十分意外。

对于信源速率小于1的情况暂且讨论到此。当$R>1$时，情况怎样呢？我们怎样进行准确的通信呢？

例如，若$R>1$，我们只能传送信源输出比特的$1/R$，而要求接收者猜测其余部分，这种猜测就象猜一枚翻转的硬币的正反面一样。对这种不太高明的方案，不难算出所产生的误码率p_e

$$\begin{aligned} P_e &= \frac{1}{R}\times p + \frac{R-1}{R}\times\frac{1}{2} \\ &= \frac{1}{2} - \left(\frac{1}{2} - p\right)/R \end{aligned} \qquad (0.4)$$

当$R>1$时，采用另一种稍好一点的方法，以$R=3$为例说明之。若$R=3$，在信道传送的比特数只是信源发出比特数的1/3，因此发送者将信源输出比特分为三个一组，且仅

传送三个一组的择多判决符号。例如，信源输出为101110101 000101，发送者在信道上传送11101。接收者只要对接收到的每一比特进行三次重复。这时，若信道干扰歪曲了第2个传输比特，接收者将收到10101，他把它扩展为 1110001110 00111，从而造成5个比特错误。一般地，由此所产生的误码率为

$$P_e=\frac{1}{4}\times(1-p)+\frac{3}{4}\times p$$

$$=\frac{1}{4}+\frac{p}{2} \qquad (0.5)$$

注意，它小于1/3 + p/3，后者是当R = 3时，我们采用原始的“硬币翻转”策略时所产生的错误概率。当R取其它整数时，这个策略的一般形式留作习题（见习题0.4）。

至此，我们所考虑的方案都是极平凡的，但并非完全无意义。下面，让我们给出一个不平凡的例子，它在1948年之前实际上还不知道。

设R = 4/7，由此可知，对信源发出的每4个比特在信道上传送时只能发送3个附加比特。我们将精心地选择这3个附加比特：若4个信源比特用x_0, x_1, x_2, x_3表示，则附加的或冗余的或一致校验比特，标记为x_4，x_5，x_6，且由如下方程决定，

$$\begin{aligned} x_4 &\equiv x_1 + x_2 + x_3 \quad (\mathrm{mod}2), \\ x_5 &\equiv x_0 + x_2 + x_3 \quad (\mathrm{mod}_2), \qquad (0.6) \\ x_6 &\equiv x_0 + x_1 + x_3 \quad (\mathrm{mod}_2)。 \end{aligned}$$

因此，若（x_0，x_1，x_2，x_3）=（0110），则（x_4，x_5，x_6）=（011），而在信道上传输的7比特码字为0110011。

为了叙述接收者如何由受到干扰而被歪曲的7比特码字作出对4个信源比特的估计，即，叙述它的**译码算法**，我们用下面的方式重新写出一致校验方程式（0.6）

$$x_1+x_2+x_3+x_4=0,$$
$$x_0+x_2+x_3+x_5=0, \qquad (0.7)$$
$$x_0+x_1+x_3+x_6=0。$$

（在（0.7）式中，运算是模2运算），用略有不同的方式来叙述，若矩阵H定义为

$$H=\begin{pmatrix}0&1&1&1&1&0&0\\1&0&1&1&0&1&0\\1&1&0&1&0&0&1\end{pmatrix},$$

可看出，16个可能的码字中的每一个$x=(x_0, x_1, x_2, x_3, x_4, x_5, x_6)$都满足矩阵矢量方程

$$Hx^T=\begin{pmatrix}0\\0\\0\end{pmatrix}。 \qquad (0.8)$$

（式中的上标T表示“转置”。）

设想对于在BSC上传输的比特按下述原则加上(mod2) 0或1：若该比特未发生错误时加0，否则加1。这种作法证明是有效的。若传输矢量为$x=(x_0, x_1, x_2, \cdots, x_6)$，则接收矢量为$y=(x_0+z_0, x_1+z_1, \cdots, x_6+z_6)$，其中$z_i=1$，若第$i$分量发生了传输错误；否则$z_i=0$。于是，若$z=(z_0, \cdots, z_6)$表示**错误模式**，则$y=x-z$。接收者只知道$y$，但他想知道$x$，这时他做一件很巧妙的事：他计算下面的矢量$S=(s_0, s_1, s_2)$：

$$S^T=Hy^T$$
$$H(x+z)^T$$

$$= Hx^T + Hz^T$$

$$= Hz^T \quad (\text{见}(0.8)) \qquad (0.9)$$

这里，s称之为y的校正子[5]，在校正子中，0表示y满足相应的一致校验方程，而1表示不满足。根据（0.9）式，校正子不取决于发送的码字，而仅与错误模式z有关。但是，因为 $x = y + z$，若接收者能求出z，他也就求得了x，因此，他把注意力集中在求z上。方程 $S^T = Hz^T$ 表明S是H与z中的1相对应的那些列的（二进制）和，即，与被信道歪曲的码字的比特相对应的那些列的和：

$$S^T = Z_0\begin{pmatrix}0\\1\\1\end{pmatrix} + Z_1\begin{pmatrix}1\\0\\1\end{pmatrix} + \cdots + Z_6\begin{pmatrix}0\\0\\1\end{pmatrix} \qquad (0.10)$$

在已计算出S的情况下，接收者的任务就是要解出关于z的方程 $S^T = HS^T$。可惜，未知元为7而只有三个方程，因此，对任一S，z有16种可能性。对于z的验前有128种可能性，这点已是明确的。但是，接收者在剩下的16个中如何选择呢？例如，设接收到的y =（0111001），则S =（101），于是，16种可供选择的z为

0100000	0010011
1100011	0001010
0000101	0111001
0110110	1010000
0101111	1001001
1000110	1111010
1110101	0011100
1101100	1011111

面对这个可能的错误模式集合，如何处理是显然的：由于原误码率$p<\frac{1}{2}$，在一个错误模式中1（错误）最少者，很可能是实际的错误模式。在上例中，我们很幸运：存在着唯一的最小重量的错误模式（0100000），它的重量恰好为1。因此，在这种情况下接收者关于z的最佳估计（基于校正子和信道转移概率矩阵）是z=(0100000)；而传输的码字的估计是x= y+ z=（0011001）；最后，得到4个信源比特的估计为（0011）。

当然，在上述例子中，我们未必真正幸运。因为可以证明，对任意校正子S，总存在唯一满足$Hz^T= S^T$的重量为0或1的解。为了说明这一点，注意到若S=（000），则z=（0000000）是所求的解。若S≠（000），则S^T必作为H的某一列出现；如果S^T= H的第i列，那么在第i个位置上是1而其它位置上是0的错误模式z，就是$Hz^T= S^T$的唯一最小重量解。

现在，我们可以正式地描述称之为（7，4）汉明码的译码算法。若接收矢量为y，接收者实行下述步骤：

1。计算校正子$S^T= Hy^T$。

2.若s=（000），令$\hat{z}=0$；转入4。

3.判明H的唯一等于S的列；称它为列i；令$\hat{z}$除第i个座标上为1外，其余全是0。

4.令$\hat{x}=y+\hat{z}$（这是译码者对传送码字的估计）

5.输出（$\hat{x}_0$，$\hat{x}_1$，$\hat{x}_2$，$\hat{x}_3$），x的前4个元素。（这是译

码者对原信源比特的估计）。

当然，由这种算法得到的$\hat{z}$可能不等于实际上的错误模式z但是，若信道引起的错误至多一个，即若z的重量为0或1，

这时由上述讨论可知$\hat{z}=z$。因此汉明码是纠单个错误的码，实际上不难看出，当且仅当信道引起的错误多于1个时，上述译码算法对正确识别原码字x无效。因此，若用P_E表分组误码率$p\{\hat{x}=x\}$则

$$P_E=\sum_{k=2}^{7}\binom{7}{k}p^k(1-p)^{7-k}$$

$$=21p^2-70p^3+\cdots。$$

当然，P_E不能说明所有的各种情况，即使$\hat{x}\neq x$，$\hat{x}$的分量中的一部分或许还是正确的。若我们用$P_E^{(i)}$表示误码率$P\{\hat{x}_i\neq x_i\}$，则可证明，当$0\leqslant i\leqslant 6$时

$$P_e^{(i)}=9p^2(1-p)^5+19p^3(1-p)^4+16p^4(1-p)^3$$
$$+12p^5(1-p)^2+7p^6(1-p)+p^7$$
$$=9p^2-26p^3+\cdots。\qquad(0.11)$$

比较（0.1）和（0.11）式，我们看到对具有很小的原误码率的BSC，以速率为4/7=0.571进行汉明编码与以速率为1/3=0.333实行原始重复编码的效果大致相同。

我们还可以通过交换发送者和接收者的地位，利用(7,4)

汉明码以R = 7/4的速率进行通信。这时，发送者把信源比特序列划分为7个一组，采取上述译码算法把每7个一组简化为只有4个比特（在这个意义上，译码算法变为“编码算法”），在信道上传输这4个比特。接收者对收到的比特加上由一致校验规则（0.6）算出的3个附加比特再进行译码。这个方案

产生的误码率 $P_e^{(i)} = p\{\hat{x}_i \neq x_i\}$ 与i无关，但平均值 $P_e = \sum_{i=1}^{6} P_e^{(i)}/7$ 为

$$P_e = \frac{1}{8}(1-p)^4 + \frac{53}{28}(1-p)^3 p + 3(1-p)^2 p^2 + \frac{59}{28}(1-p)p^3 + \frac{7}{8}p^4$$

$$= \frac{1}{8} + \frac{39}{28}p + \cdots。 \qquad (0.12)$$

当BSC无干扰（p = 0）时，它比R = 7/4时的“抛硬币”技术要好，后者由（0.4）式给出Pe = 3/4 = 0.214。

至此，我们通过一个具体的BSC（如p = 0.1）总结一下前面的讨论，同时对所讨论的每一通信方案，在（x、y）平面上标出一个点，这里的x = R为速率，y = Pe为误码率，如图0.1所示。若继续发挥创造性，我们可以继续构造具体的方案，同时在图上标出这些对应的点。当然，我们的最终目的是研究哪些点可实现，哪些点不能实现。令人难以置信的是，仙农已达到了这一目标。但在给出仙农的结果之前，让我们把速率为R，误码率为Pe的编码方案的概念格式化。

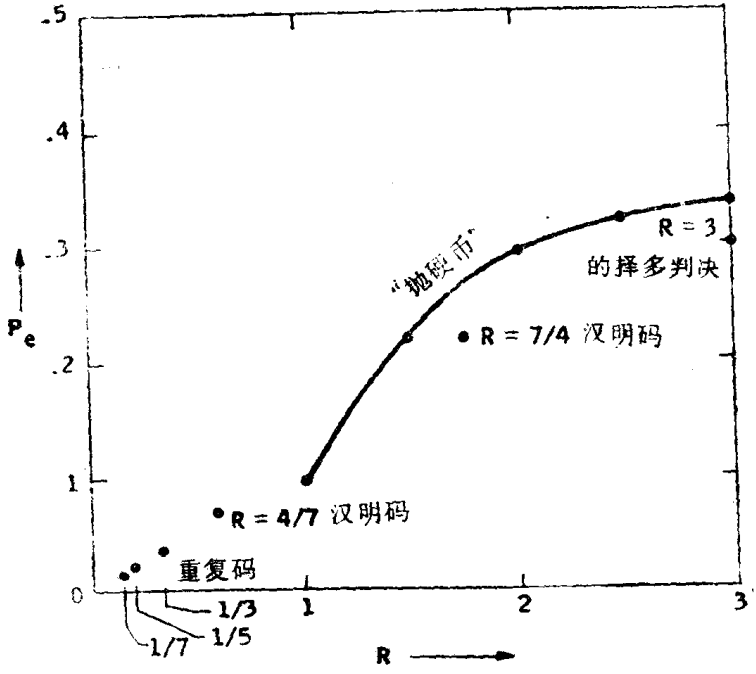

图0.1 概率p=0.1的BSC的一些可实现的（R，Pe）对。

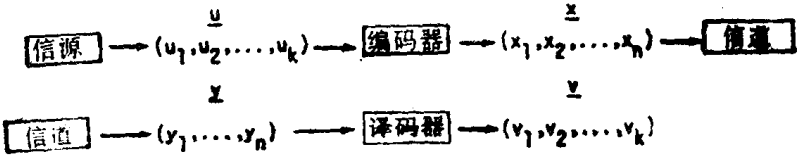

图0.2 二元对称信源和信道的一个(n，k)码

如图0.2所设定的，一个（n，k）码是一种方案，其中信源序列被分为若干组，每组k比特，而且每一k比特信源

u组被变换（编码）为一个n比特的码字x，它在信道上被传输和接收，可能被歪曲成y。译码者把n比特码字y变换成一个k比特的v，它是对原信源序列u的估值。这个通信系统的传输速率$R=k/n$，其误码率定义为

$$P_e=\frac{1}{k}\sum_{i=1}^{k}P_e^{(i)}$$

这里

$$P_e^{(i)}=P\{v_i\neq u_i\},\quad i=1,2,\cdots,k。$$

（到目前为止所述的每一方案都合于这种描述，也许$R\geqslant 1$的“抛硬币”策略例外，见习题0.5。）若存在一个(n,k)码，满足$k/n\geqslant x$，$P_e\leqslant y$，我们说图0.1中的一个点(x,y)是“可实现的”。这问题已解决，图0.3给出了一个确定的BSC（$p=0.1$）可实现点的集合。当然，搞清楚图0.3的关键是说明可实现与不可实现区域的界限。为了给出这种说明，我们需要引入重要的二进制熵函数：

$$H_2(x)=-x\log_2 x-(1-x)\log_2(1-x),\quad 0<x<1,$$
$$H_2(0)=H_2(1)=0。\qquad(0.13)$$

图0.4示出了$y=H_2(x)$的曲线。（$H_2(x)$的一些重要的特点在习题0.10中给出。）我们现在来描述图0.3中可实现与不可实现区域之间的界限。界的曲线部分是满足下式的(R,P_e)点集：

$$R=\frac{1-H_2(0.1)}{1-H_2(p_e)},\quad 0\leqslant P_e\leqslant\tfrac{1}{2}。\qquad(0.14)$$

界的其余部分是R轴上从$R=0$到$R=1-H_2(0.1)=0.531$的线段。对一个一般的BSC，情况完全一样，只要用