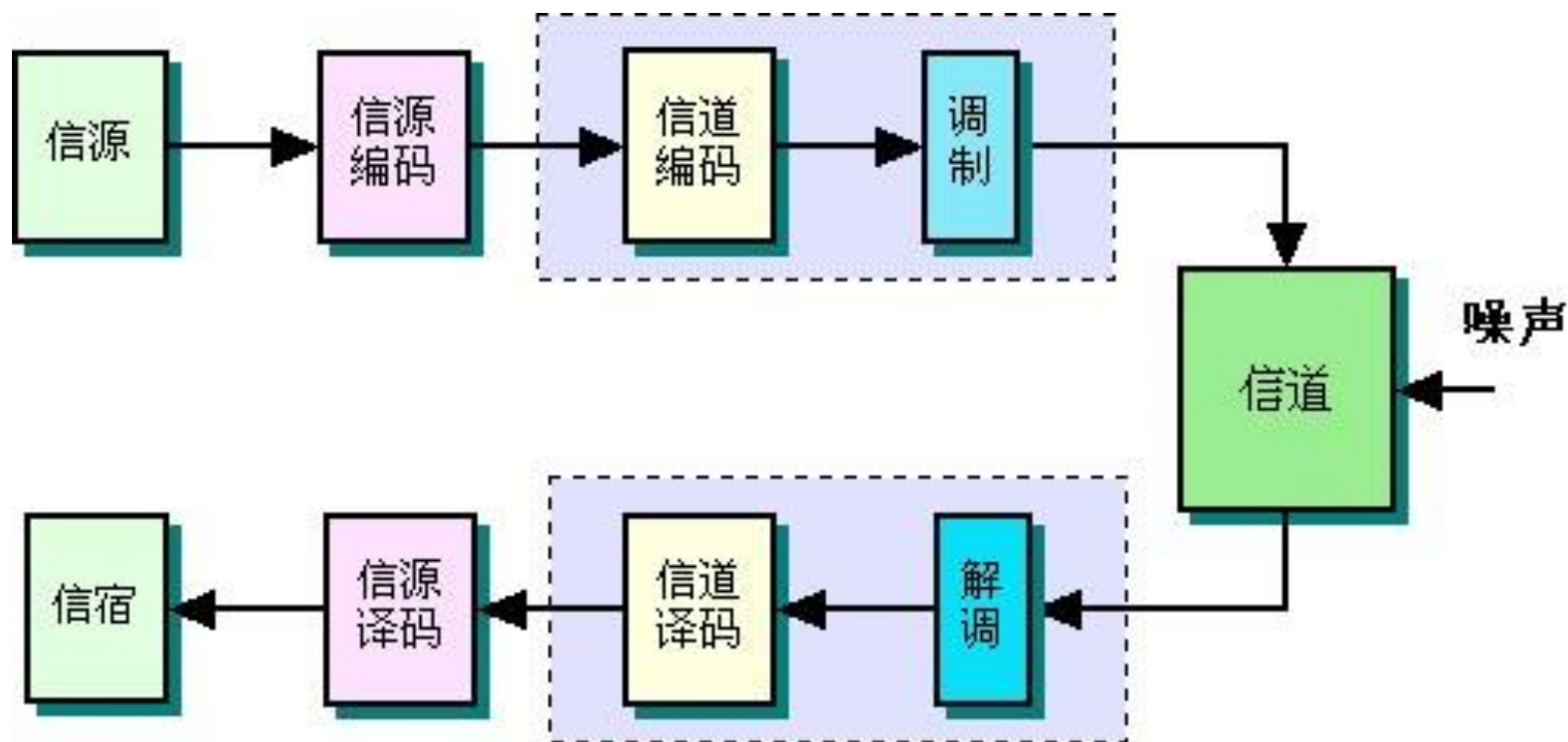


15. 其他差错控制编码技术

- TCM技术
- 交织技术
- 级联码
- 差错检测和ARQ技术

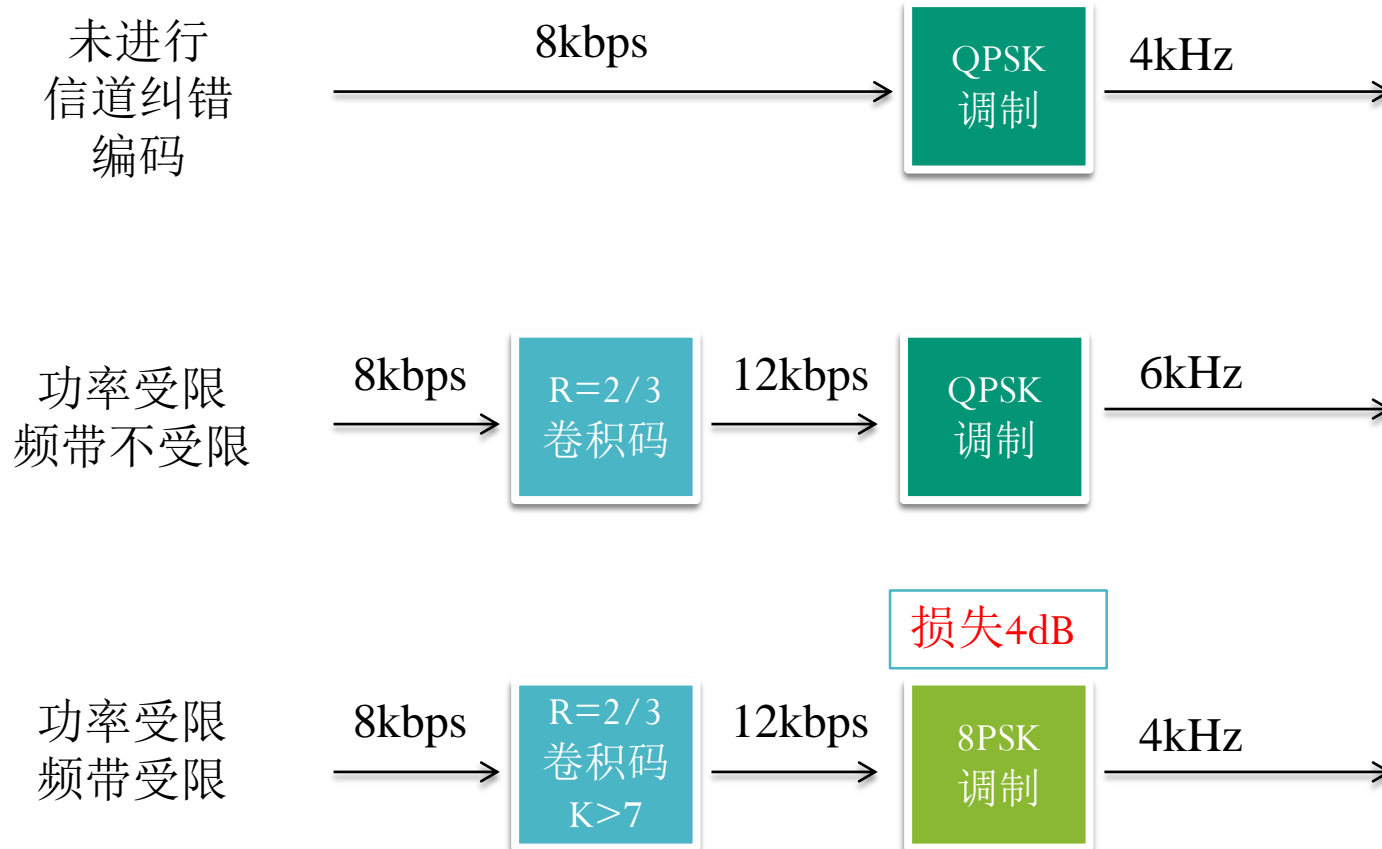
- ◆ 引言
- ◆ 调制信号空间的分解
- ◆ TCM编码
- ◆ TCM译码
- ◆ 性能分析
- ◆ 应用

- TCM(Trellis-Coded Modulation)是一种将信道编码和调制相结合的技术



- 在采用相同调制信号的前提下， (n, k) 分组码和 $(K, R=k/n)$ 卷积码都是通过扩展信道带宽 $(1/R)$ 倍) 来获得编码增益，适合于功率受限而频带不受限的系统
- 在频带受限系统中，为了获得编码增益，即节省功率，必须采用不同的调制信号集

TCM-引言



- 如系统中信号未编码时采用QPSK调制信号传输
- 经过 $R=2/3$ 卷积码编码后，为不增加信道带宽，则需采用8PSK调制信号来传输
- 而8PSK信号为了获得与QPSK信号相同的误码率，功率需增加4dB
- 因此要求卷积码的编码增益超过 4dB, 才能节省功率
- 这就要求采用长约束长度（如 $K>7$ ）卷积码，因而增加了设备的复杂度

- TCM技术通过将**编码**和**调制**技术相结合，在不增加信道带宽的前提下，获得显著的编码增益
 - 简单的4状态TCM可获得**3dB**的编码增益
 - 复杂的TCM可获得**6dB**,甚至更高的编码增益
- 这些增益是在**不增加信道带宽**或**降低信息传输速率**的前提下得到的

- TCM技术特别适合**频带**和**功率**同时**受限**的通信系统，如卫星通信系统
- TCM技术最早在1976年由Ungerboeck提出
- 目前对TCM技术的理论研究和实际应用得到迅速发展

G. David Forney, Jr., Gottfried Ungerboeck, “Modulation and Coding for Linear Gaussian Channels”, IEEE Transactions on Information Theory, Vol.44, No. 6, October 1998.

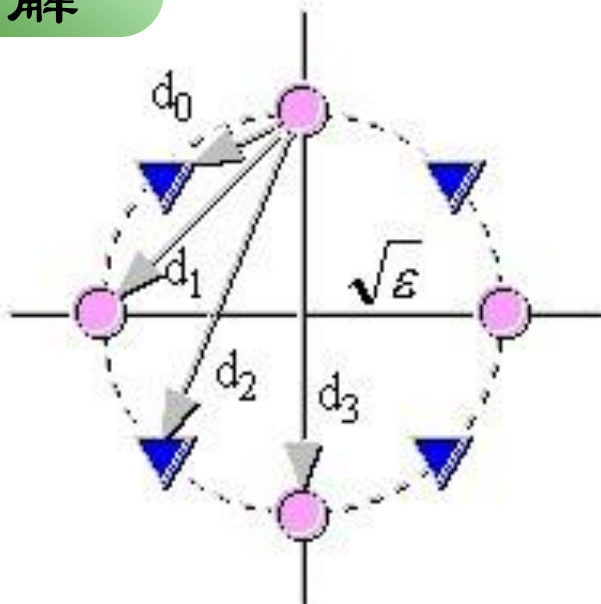
- ◆ 引言
- ◆ 调制信号空间的分解
- ◆ TCM编码
- ◆ TCM译码
- ◆ 性能分析
- ◆ 应用

TCM-调制信号空间的分解

- 所谓调制信号空间的分解，是将信号空间的调制信号与二进制序列一一对应起来
- **基本原则：每次分解，都使子集中信号之间的欧几里德距离尽可能大！**
- TCM技术常用的调制信号有PAM, PSK, DPSK和QAM信号
- 下面以8PSK和16QAM调制信号为例，介绍调制信号空间的分解，即**星座图分解**

TCM-调制信号空间的分解

8PSK
信号空间
分解



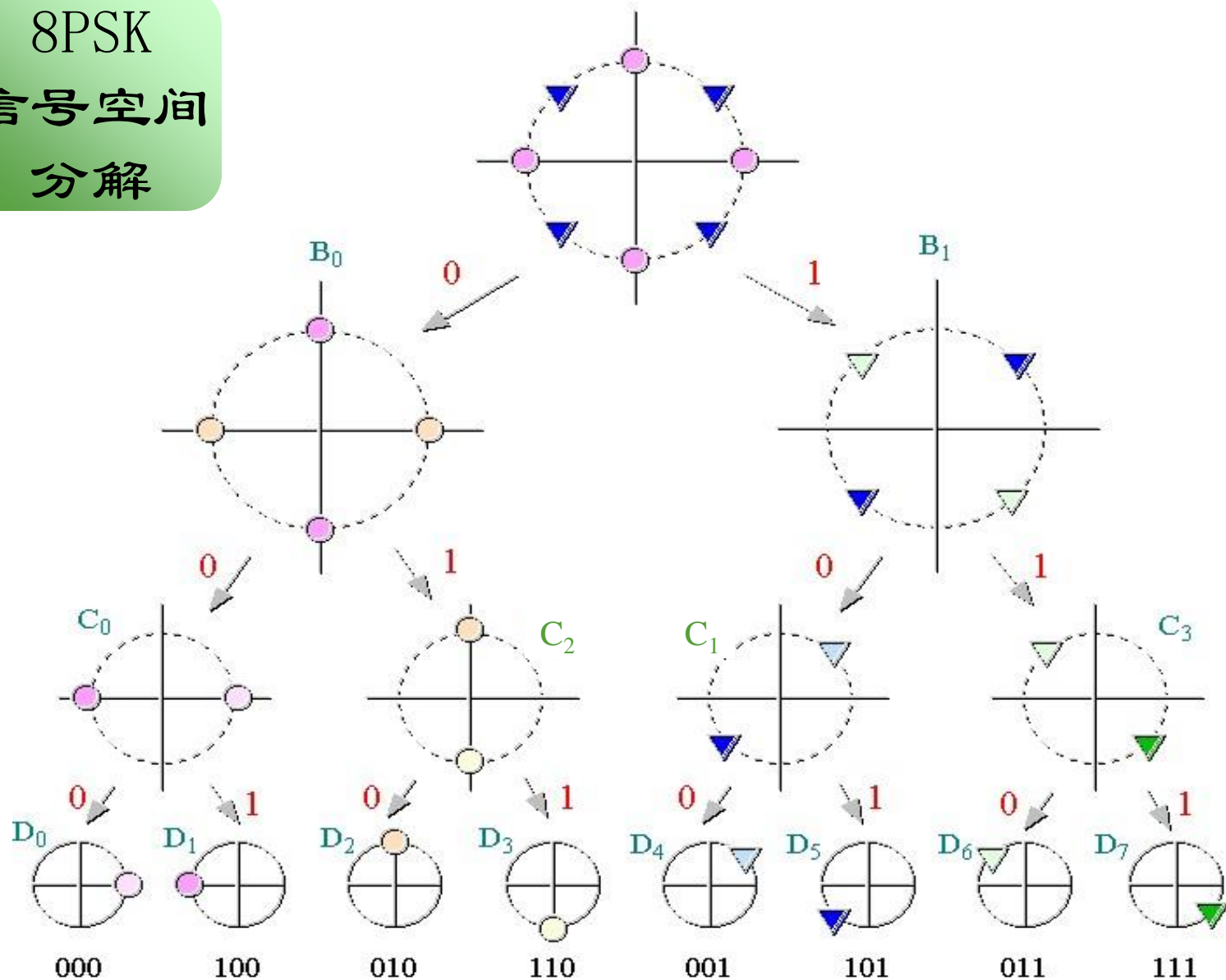
$$d_0 = \sqrt{(2 - \sqrt{2})\epsilon}$$

$$d_1 = \sqrt{2\epsilon}$$

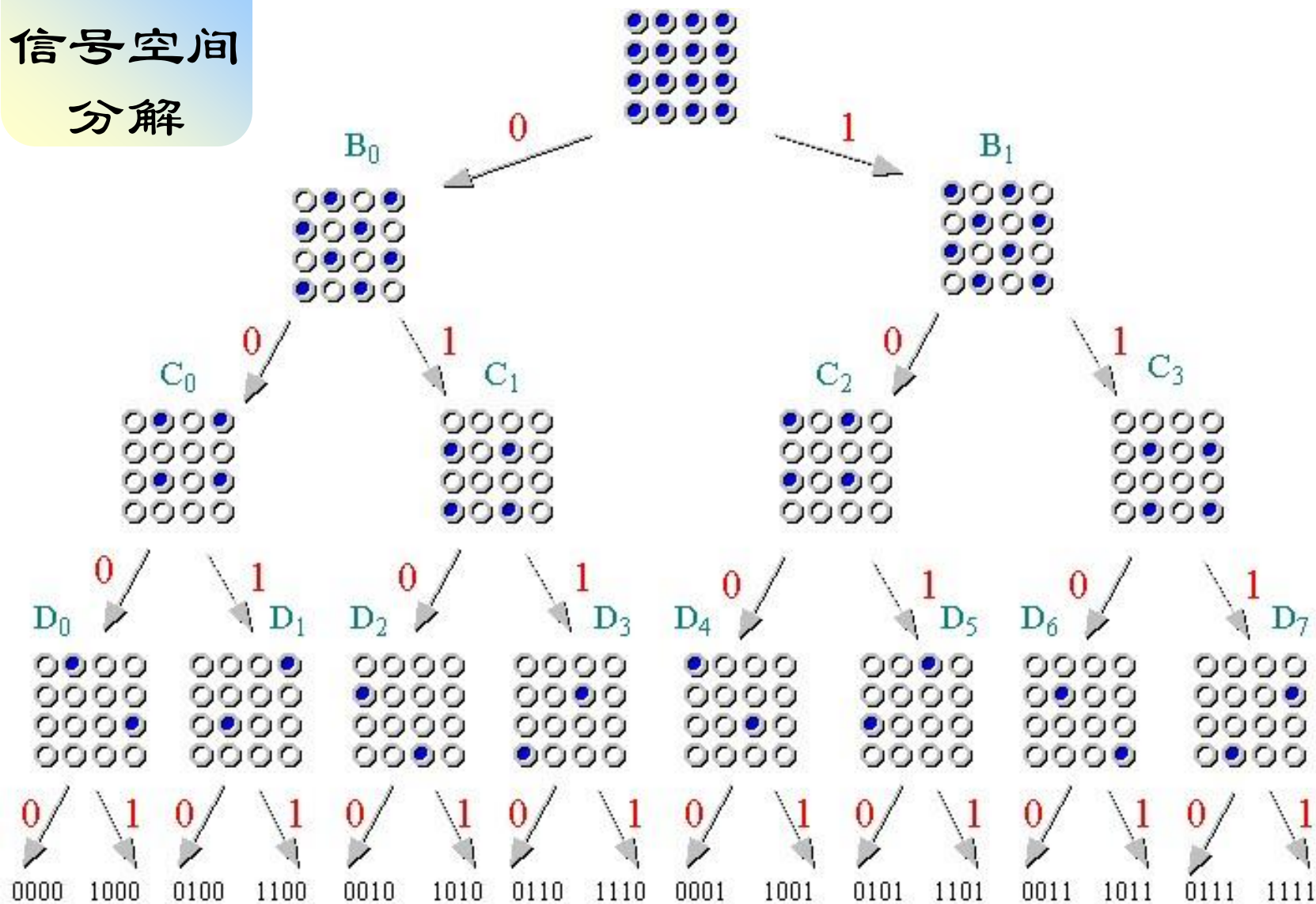
$$d_2 = \sqrt{(2 + \sqrt{2})\epsilon}$$

$$d_3 = 2\sqrt{\epsilon}$$

8PSK
信号空间
分解

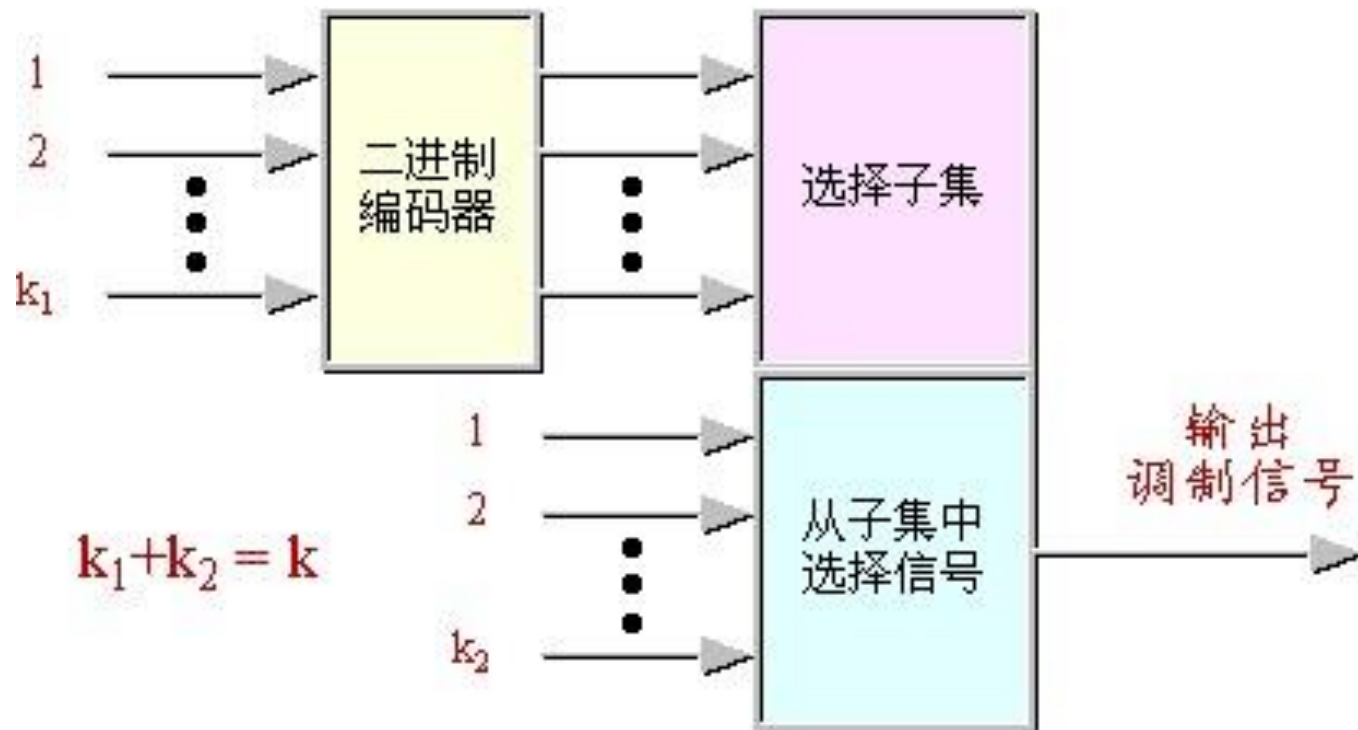


16QAM 信号空间 分解

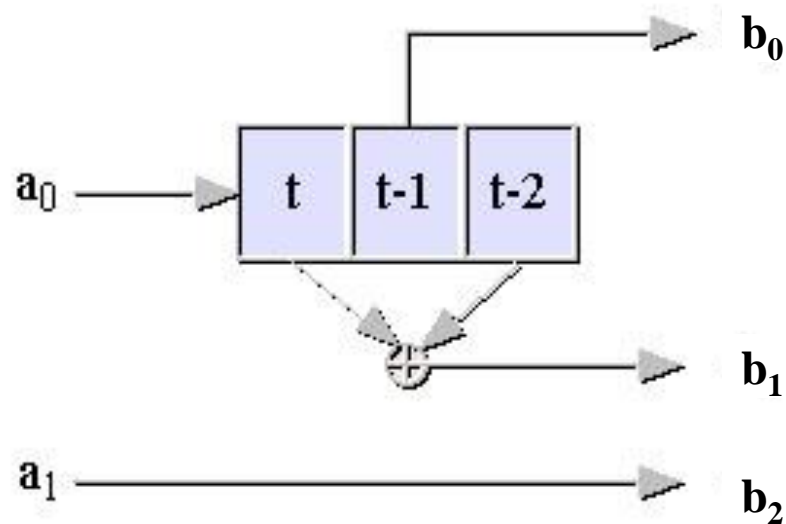
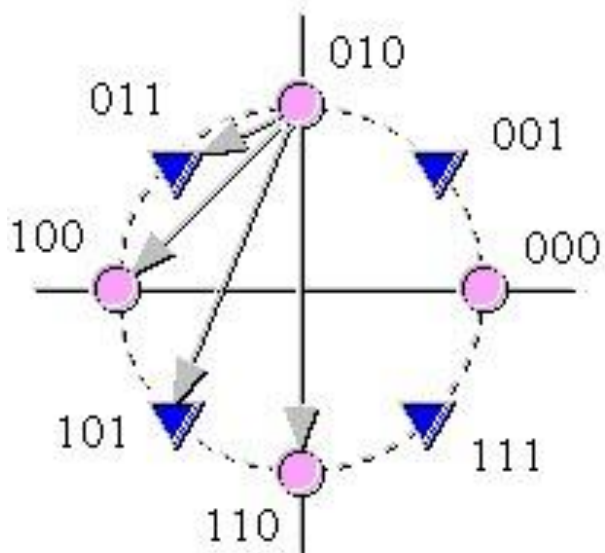


- ◆ 引言
- ◆ 调制信号空间的分解
- ◆ *TCM编码*
- ◆ TCM译码
- ◆ 性能分析
- ◆ 应用

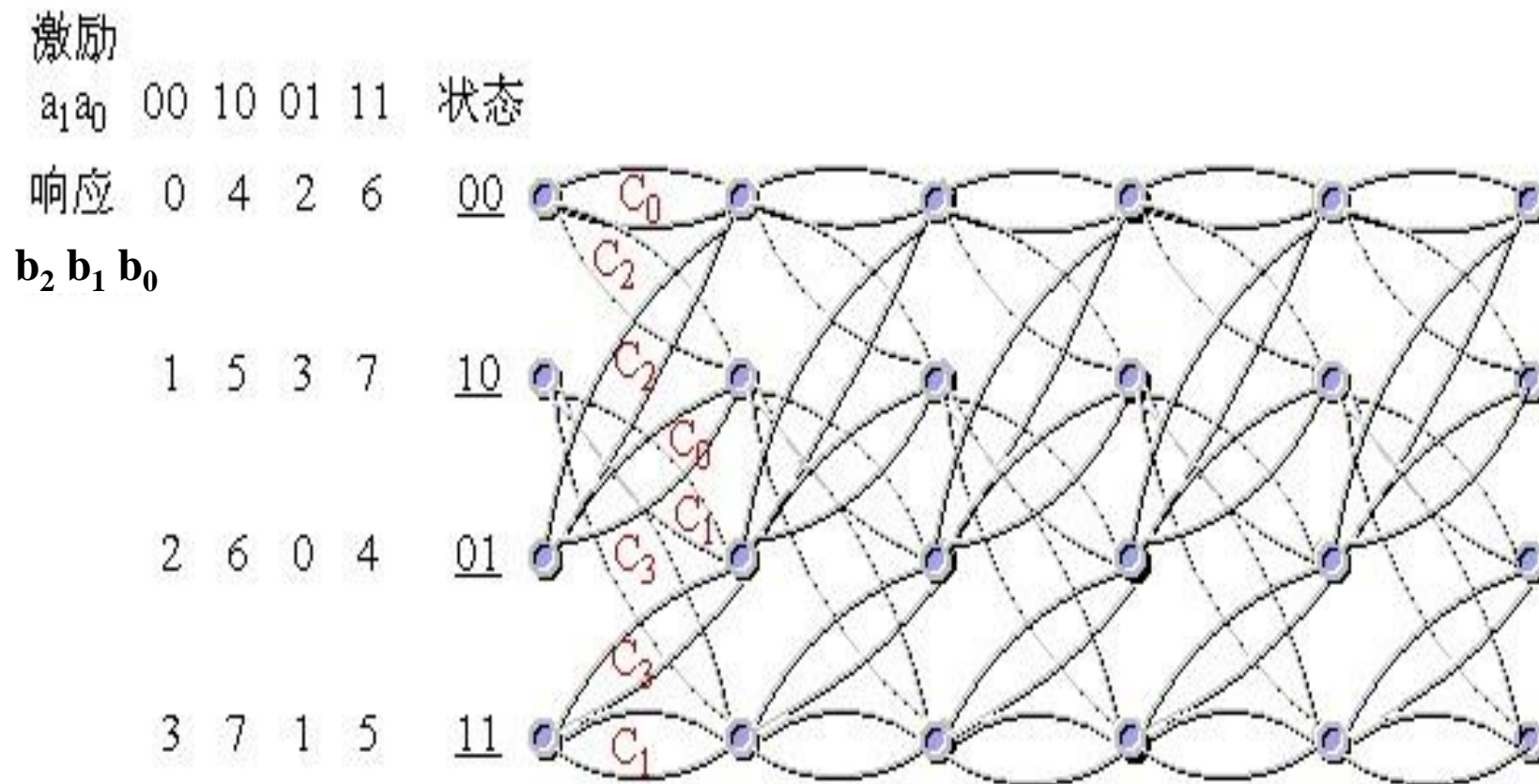
- 编码器原理图



- 例15.1：一种4状态-8PSK TCM编码器



- 例15.1:(续)网格图为



- 例15.1:(续)编码原则

- 平行分支具有最大的欧几里德距离:

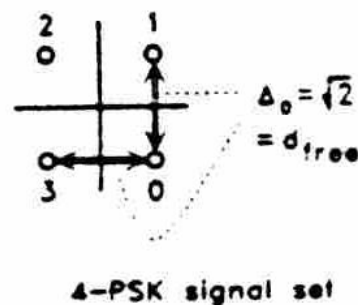
$$d_3 = 2\sqrt{\varepsilon}$$

- 从任一状态出发，或回到同一状态的子集具有最大的欧几里德距离

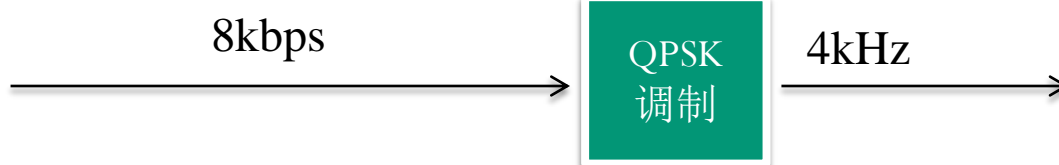
$$(C_0, C_2) \quad (C_1, C_3)$$

- 所有信号使用的概率相同

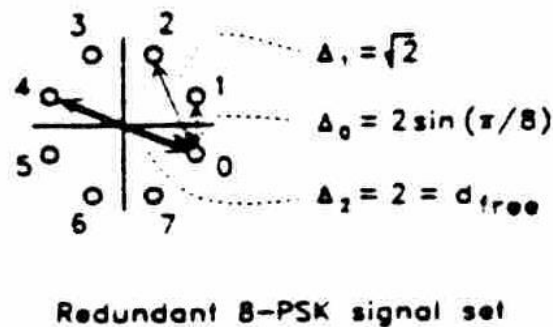
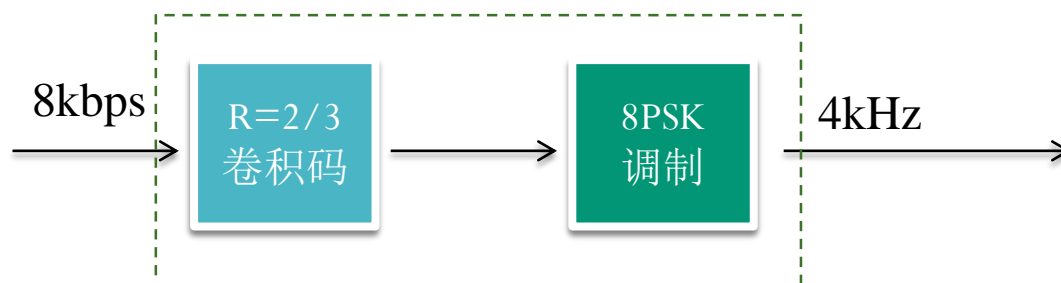
对比未编码QPSK



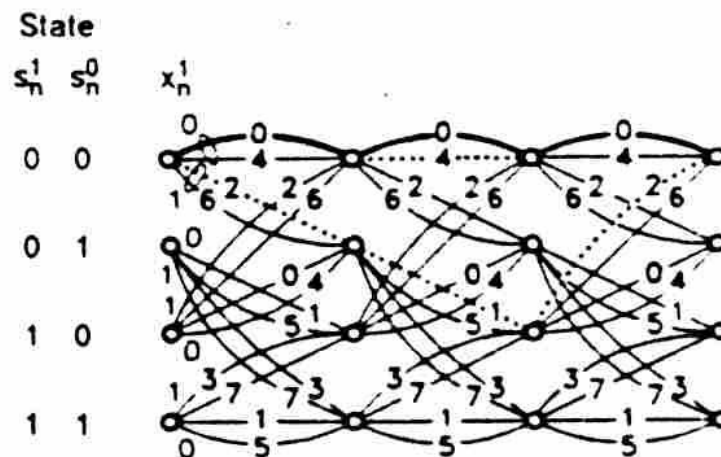
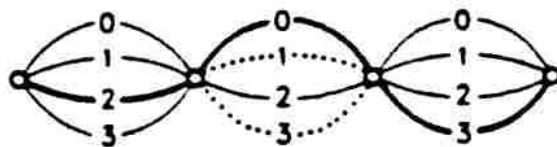
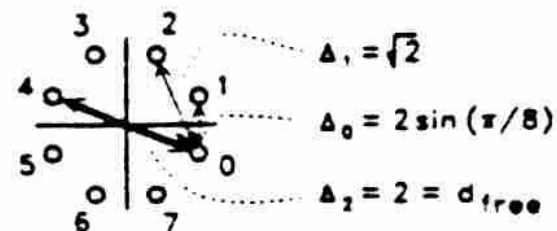
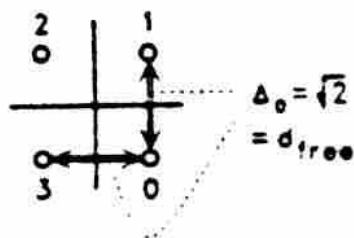
未进行
信道纠错
编码



功率受限
频带受限



TCM-编码



- 例15.1:(续)对比未编码QPSK

- 网格图上，未编码QPSK信号所对应路径间的最小欧几里德自由距离为

$$d_{free1} = \sqrt{2\varepsilon}$$

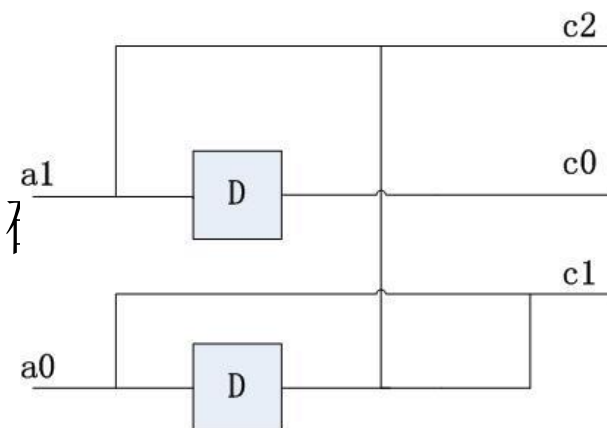
- 经过TCM编码后的4状态-8PSK信号所对应路径间的最小欧几里德自由距离为

$$d_{free2} = 2\sqrt{\varepsilon}$$

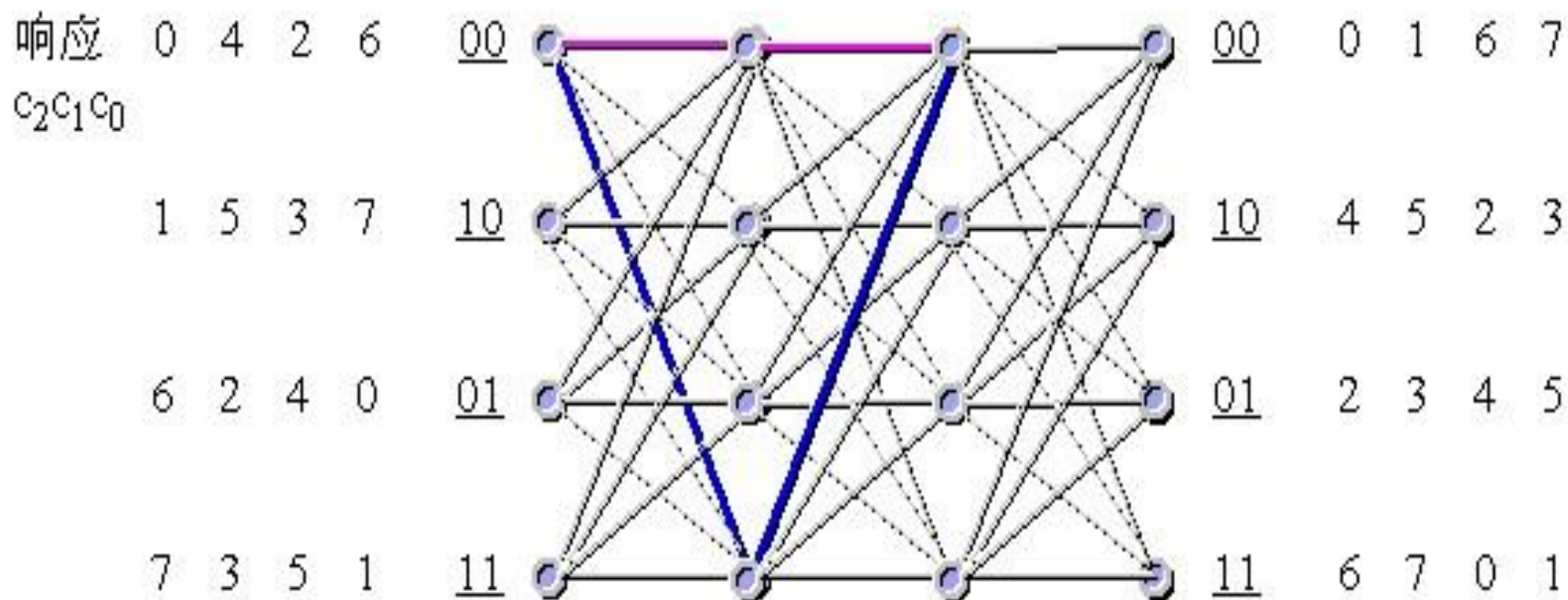
- 编码增益为

$$\Delta = 10\lg\left(\frac{d_{free2}^2}{d_{free1}^2}\right) = 10\lg\left(\frac{4\varepsilon}{2\varepsilon}\right) = 3.01(dB)$$

- 例15.2: 另一种4状态-8PSK TCM编码图:



激励
 $a_1 a_0$ 00 10 01 11 状态

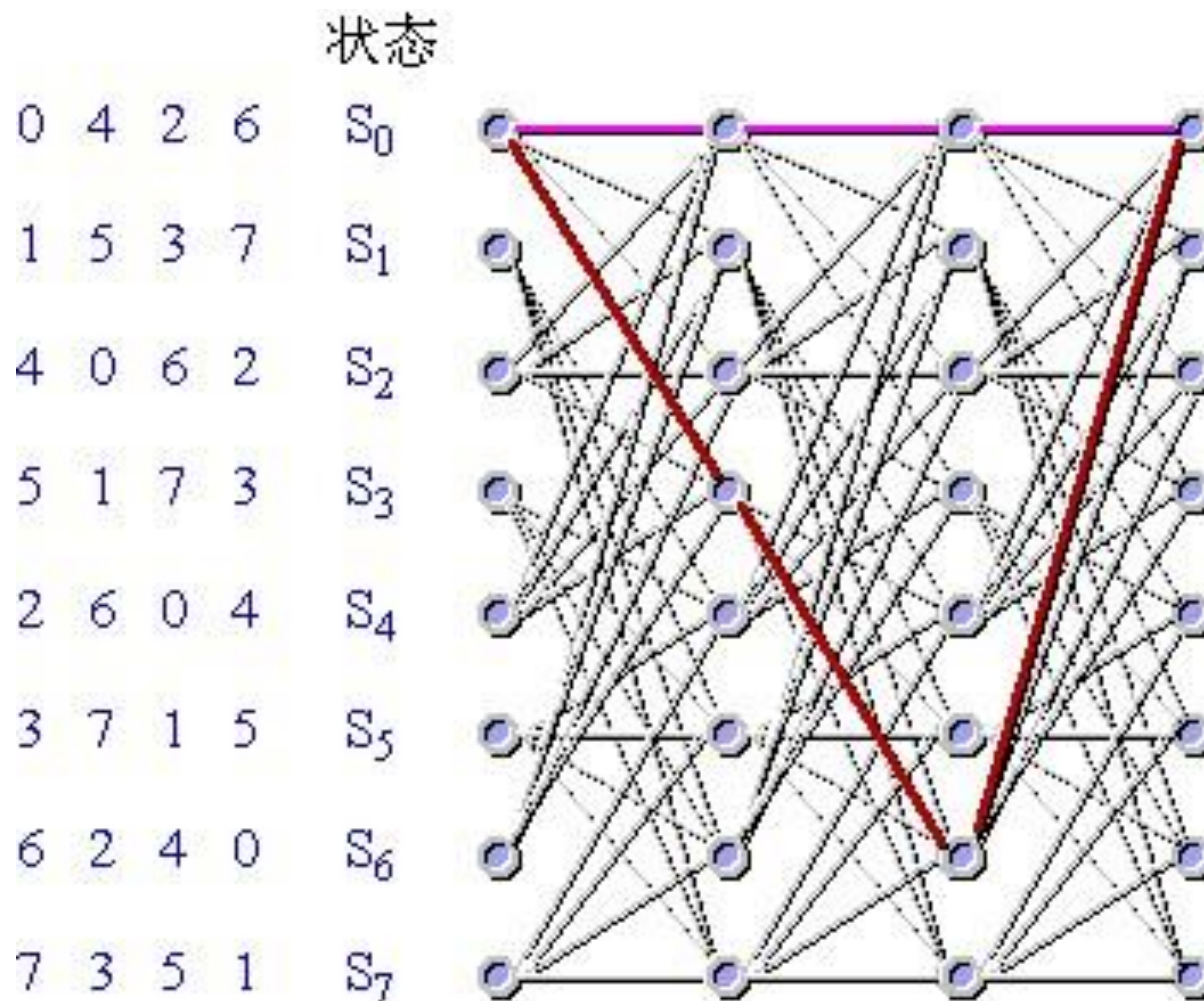


- 例15.2(续):
 - 图中与全零路径间具有最小欧几里德距离的路径已标出，状态变化过程为 $s_0 \rightarrow s_3 \rightarrow s_0$ ，

$$d_{free} = \sqrt{d_{0,6}^2 + d_{0,7}^2} = \sqrt{(4 - \sqrt{2})\varepsilon}$$

– 该TCM编码的最小欧几里德距离比例 15.1中的小，其性能不如例15.1中的TCM编码！

- 例15.3:一种8状态-8PSK TCM编码所对应的网格图:



- 例15.3(续):
 - 图中与全零路径间具有最小欧几里德距离的路径已标出,
状态变化过程为 $s_0 \rightarrow s_3 \rightarrow s_6 \rightarrow s_0$,

$$d_{free} = \sqrt{d_{0,6}^2 + d_{0,7}^2 + d_{0,6}^2} = \sqrt{(6 - \sqrt{2})\varepsilon}$$

–该TCM编码的最小欧几里德距离比例 15.1中的大, 其性能比例15.1中的TCM编码好!

–对比未编码QPSK信号, 其编码增益为

$$\Delta = 10\lg\left(\frac{(6 - \sqrt{2})\varepsilon}{2\varepsilon}\right) = 3.6(dB)$$

- 表1--表3列出了采用不同调制信号的TCM的编码增益与编码器状态数及编码速率之间的关系
- 其编码器可采用反馈或非反馈卷积码编码器
- 可见，简单的4状态TCM与未编码调制相比，就可以获得约3dB的编码增益
- 复杂的TCM可以获得6dB甚至更高的编码增益

表 1 PAM 调制 TCM 的编码增益

编码器 状态数	k_1	编码速率 k_1/k_1+1	m=1	m=2
			4-PAM 对比 未编码 2-PAM 编码增益	8-PAM 对比 未编码 4-PAM 编码增益
4	1	1/2	2.55	3.31
8	1	1/2	3.01	3.77
16	1	1/2	3.42	4.18
32	1	1/2	4.15	4.91
64	1	1/2	4.47	5.23
128	1	1/2	5.05	5.81

源: Ungerboeck (1987)

表 8 PSK 调制 TCM 的编码增益

编码器 状态数	k_1	编码速率 k_1/k_1+1	m=2	m=3
			8-PSK 对比 未编码 4-PSK 编码增益	16-PSK 对比 未编码 8-PSK 编码增益
4	1	1/2	3.01	3.54
8	1	1/2	3.60	4.01
16	1	1/2	4.13	4.44
32	1	1/2	4.59	5.13
64	1	1/2	5.01	5.33
128	1	1/2	5.17	5.33
256	2	2/3	5.75	5.51

源: Ungerboeck (1987)

表 9 QAM 调制 TCM 的编码增益

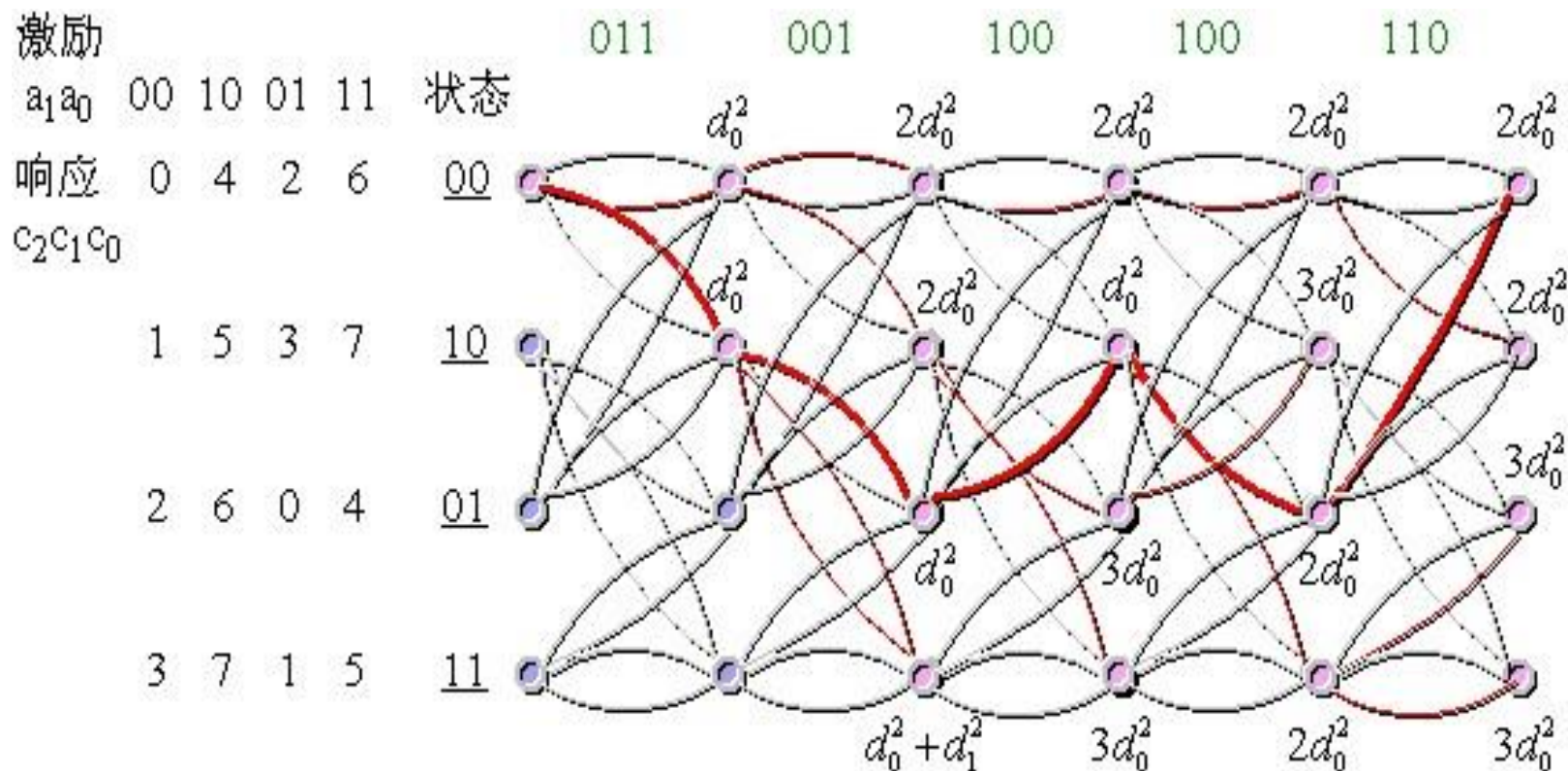
编码器 状态数	k_1	编码速率 k_1/k_1+1	m=4	m=5
			32-QAM 对比 未编码 16-QAM 编码增益	64-QAM 对比 未编码 32-QAM 编码增益
4	1	1/2	3.01	2.80
8	2	2/3	3.98	3.77
16	2	2/3	4.77	4.56
32	2	2/3	4.77	4.56
64	2	2/3	5.44	5.23
128	2	2/3	6.02	5.81
256	2	2/3	6.02	5.81

源: Ungerboeck (1987)5.81

- ◆ 引言
- ◆ 调制信号空间的分解
- ◆ TCM编码
- ◆ *TCM译码*
- ◆ 性能分析
- ◆ 应用

- TCM的译码可通过**软判决**或**硬判决**Viterbi译码器实现
- 其译码过程如下：
 - 由于在TCM网格图上的每一个**分支**都对应一个**调制信号**，因此首先需寻找一个最佳信号点，使其与此时收到的调制信号之间的欧几里德距离最小
 - 然后将选出的信号点及它与收到的调制信号之间的**平方欧几里德距离**测度用于Viterbi译码算法
 - 在网格图上找出与接收调制信号序列之间平方欧几里德距离和最小的信号路径，作为译码输出

- 例17.4：采用例17.1中的TCM编码，若接收端收到的信号 $\mathbf{r} = (011 \ 001 \ 100 \ 100 \ 110)$ ，其Viterbi译码过程如下：



- ◆ 引言
- ◆ 调制信号空间的分解
- ◆ TCM编码
- ◆ TCM译码
- ◆ 性能分析
- ◆ 应用

- 在加性高斯白噪声信道中，高信噪比条件下，TCM的误码率可表示为：

$$P_e \approx N_{free} Q\left(\sqrt{\frac{d_{free}^2}{2N_0}}\right)$$

其中 N_{free} 表示具有最小自由距离 d_{free} 的信号序列的数目，而

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp\left(-\frac{y^2}{2}\right) dy$$

TCM-性能分析

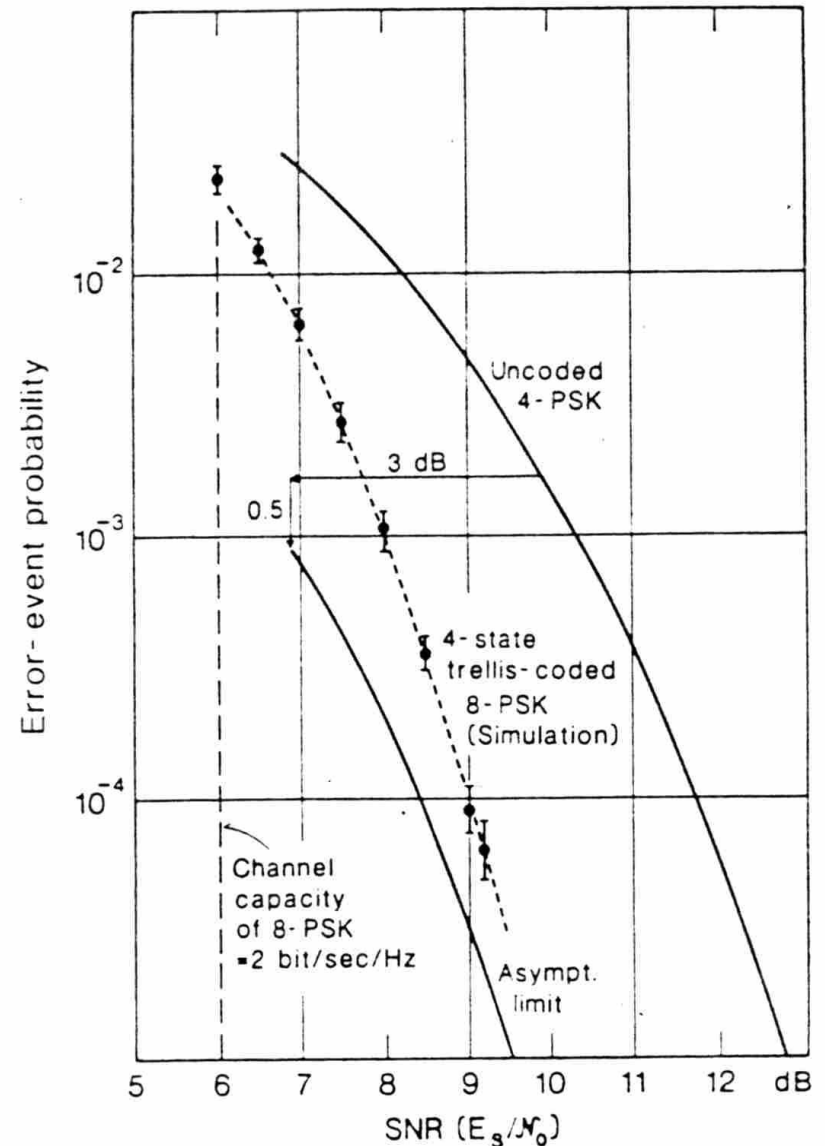
- QPSK

$$P_e \approx 2Q\left(\sqrt{\frac{2\varepsilon}{2N_0}}\right) = 2Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

- 4状态8PSK TCM

$$P_e \approx Q\left(\sqrt{\frac{4\varepsilon}{2N_0}}\right) = Q\left(2\sqrt{\frac{E_b}{N_0}}\right)$$

- QPSK曲线向左移 3dB,
向下移0.5



- ◆ 引言
- ◆ 调制信号空间的分解
- ◆ TCM编码
- ◆ TCM译码
- ◆ 性能分析
- ◆ 应用

- Wei在1984年提出的采用32QAM调制信号的非线性8状态TCM方案，被电话线调制器的国际标准V.32和V.33采用。在1987年提出的一种16状态4-维线性TCM方案，被V.34标准采用
- 美国NASA为将多维TCM技术应用于卫星移动通信系统做了大量研究工作

- DVB-S2标准中采用的ACM（Adaptive Coding and Modulation）技术也是一种将编码和调制相结合的技术
 - （BCH+LDPC）+APSK

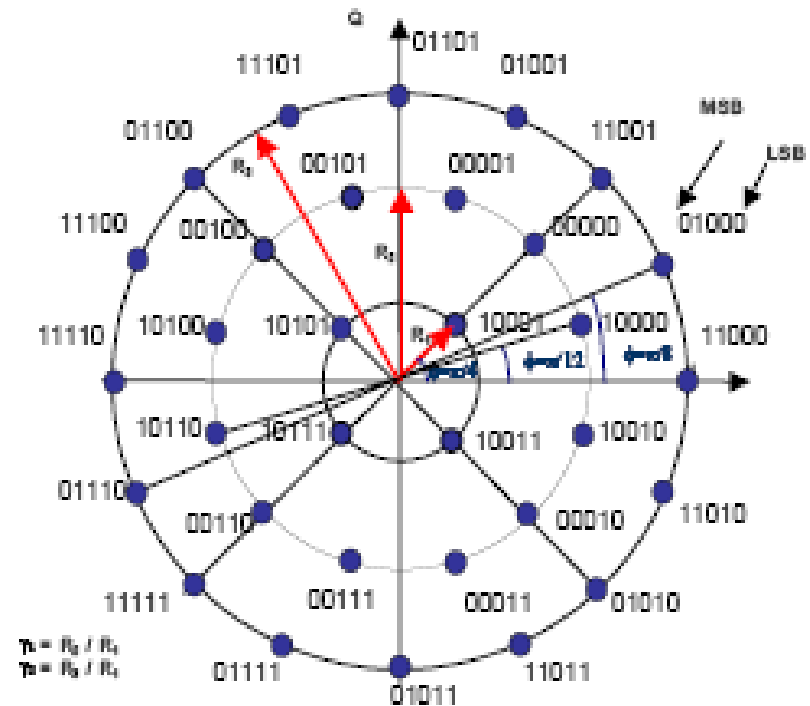
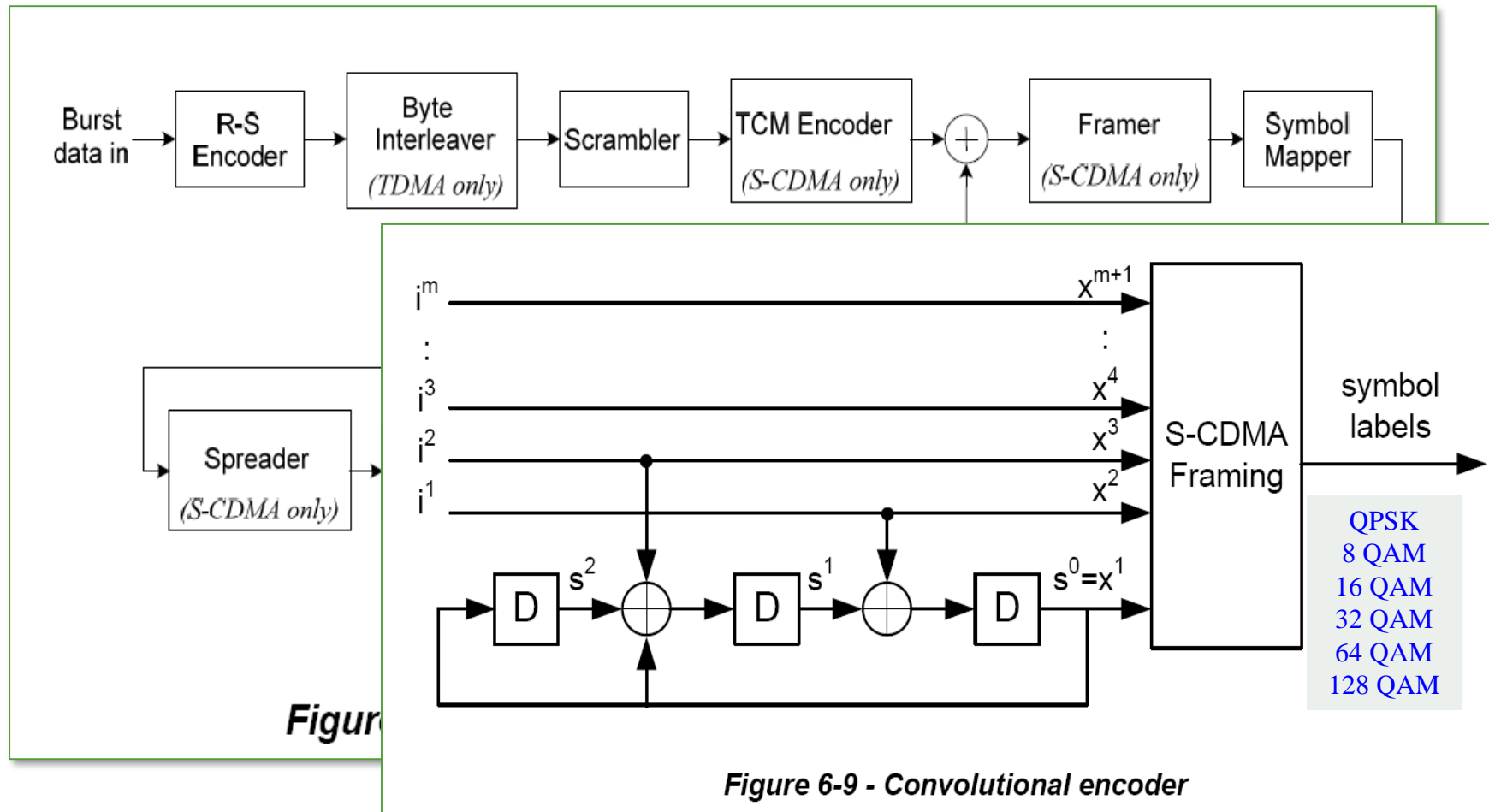


Figure 12: 32APSK signal constellation

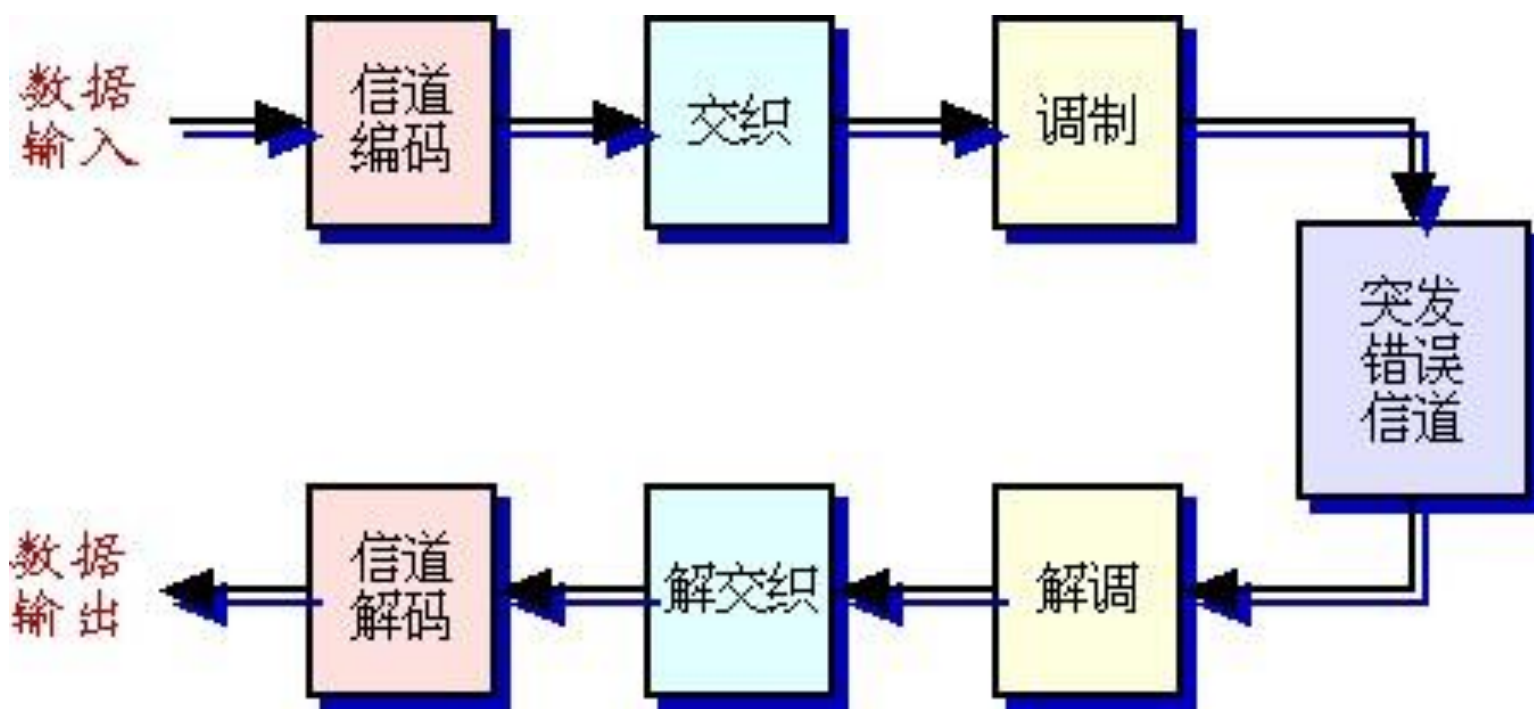
- DOCSIS3.0标准: Data Over Cable Service Interface Specifications



- TCM技术
- 交织技术
- 级联码
- 差错检测和ARQ技术

- 大部分信道纠错码型，适于纠正加性高斯白噪声信道中的随机错误
- 但在移动通信和数据存储的过程中，经常出现连续错误、突发性错误
 - 例如在多径衰落信道中，由于时变多径分集引起的信号衰落，经常使信号幅度低于噪声幅度，产生大量连续、突发错误
- **交织技术**可以把一片突发错误变为分散的随机错误，是纠正突发错误的一个有效方法，在移动通信系统中被广泛采用

- 采用交织技术的通信系统方框图



- 交织是在发送端将经信道编码的数据重新排序，调制后在信道中传输

在接收端，经解调后的数据先送入解交织器中恢复原来的顺序，然后再进行信道解码

- 交织会带来额外的时延
- 交织类型主要有方块交织和卷积交织

- 方块交织

- 将经信道编码后的数据写入 n 行 m 列的方块交织矩阵中，一般取 n 等于纠错信道码的长度，而 m 称为**交织深度**
- 写入时按列写入，读出时按行读出
- 数据在信道中也按这个顺序传输，在接收端再通过解交织器恢复原来的顺序
- 如果在传输时出现连续 mt 个突发错误，这些突发错误就会被均匀分散到 m 列中，而每一列（对应一个纠错码字）只有 t 个错误，可以用纠错能力为 t 的码型来完成纠错

- 方块交织（续）

表 8×6 方块交织

1	9	17	25	33	41
2	10	18	26	34	42
3	11	19	27	35	43
4	12	20	28	36	44
5	13	21	29	37	45
6	14	22	30	38	46
7	15	23	31	39	47
8	16	24	32	40	48

- 改进的方块交织
 - 为了使连续突发错误经过交织和解交织后的分布更加随机，交织器在输出每一行的数据时可以采用随机的次序，而在选择输出行时也可以按照随机的次序选取，甚至可以采用更加随机的交织方式
 - 例如，在TIA/EIA/IS-95-A建议中，前向信道中的数据经过16行×8列交织前后的顺序如下

交织技术

表 16×8 方块交织前的顺序（按列写入）

	000	001	010	011	100	101	110	111
0000	1	9	17	25	33	41	49	57
0001	1	9	17	25	33	41	49	57
0010	2	10	18	26	34	42	50	58
0011	2	10	18	26	34	42	50	58
0100	3	11	19	27	35	43	51	59
0101	3	11	19	27	35	43	51	59
0110	4	12	20	28	36	44	52	60
0111	4	12	20	28	36	44	52	60
1000	5	13	21	29	37	45	53	61
1001	5	13	21	29	37	45	53	61
1010	6	14	22	30	38	46	54	62
1011	6	14	22	30	38	46	54	62
1100	7	15	23	31	39	47	55	63
1101	7	15	23	31	39	47	55	63
1110	8	16	24	32	40	48	56	64
1111	8	16	24	32	40	48	56	64

交织技术

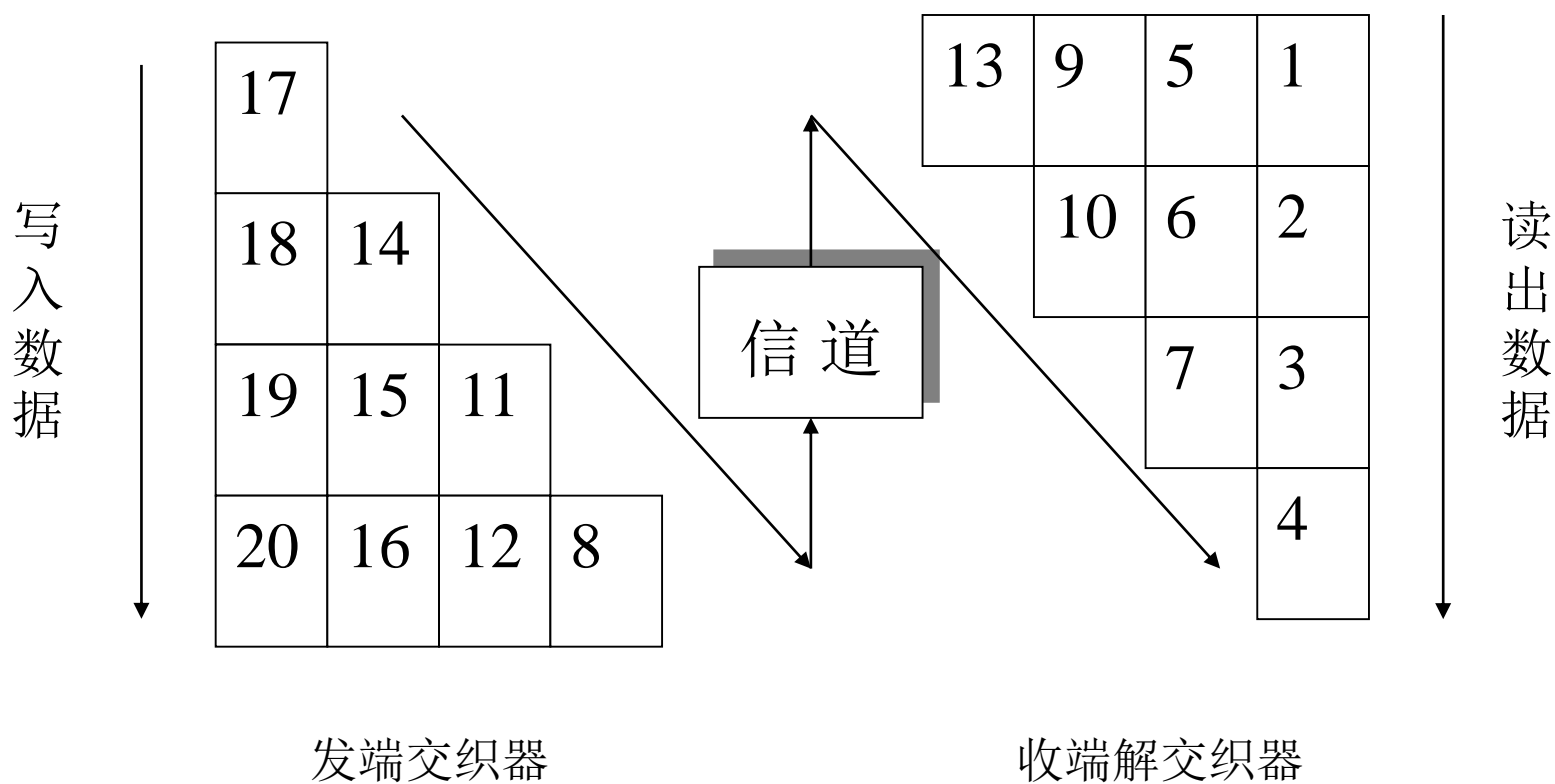
表 16×8 方块交织后的顺序（按列读出）

	0000	0100	0010	0110	0001	0101	0011	0111
	1000	1100	1010	1110	1001	1101	1011	1111
000	1	3	2	4	1	3	2	4
100	33	35	34	36	33	35	34	36
010	17	19	18	20	17	19	18	20
110	49	51	50	52	49	51	50	52
001	9	11	10	12	9	11	10	12
101	41	43	42	44	41	43	42	44
011	25	27	26	28	25	27	26	28
111	57	59	58	60	57	59	58	60
000	5	7	6	8	5	7	6	8
100	37	39	38	40	37	39	38	40
010	21	23	22	24	21	23	22	24
110	53	55	54	56	53	55	54	56
001	13	15	14	16	13	15	14	16
101	45	47	46	48	45	47	46	48
011	29	31	30	32	29	31	30	32
111	61	63	62	64	61	63	62	64

- 卷积交织

- 卷积交织又称为三角交织，由Ramsey和Forney分别在1970年和1971年提出
- 其基本原理与方块交织相同，但时延减半
- 卷积交织由于其结构特点，更适合于对卷积码编码数据进行交织
- 以 4×4 卷积交织为例，说明其工作原理：

- 卷积交织（续）



- 卷积交织（续）

- 将经信道编码后的符号按列写入发端交织器各行
- 传输时顺序地发送每行的一个编码符号，不过写入各行的符号并不立即发送，而是依行的顺序增加一个符号时延后再发送
- 接收时依次在解交织器的每行写入一个符号，经过相应的时延后再按列读出
- 上图中信道中符号传输顺序为：

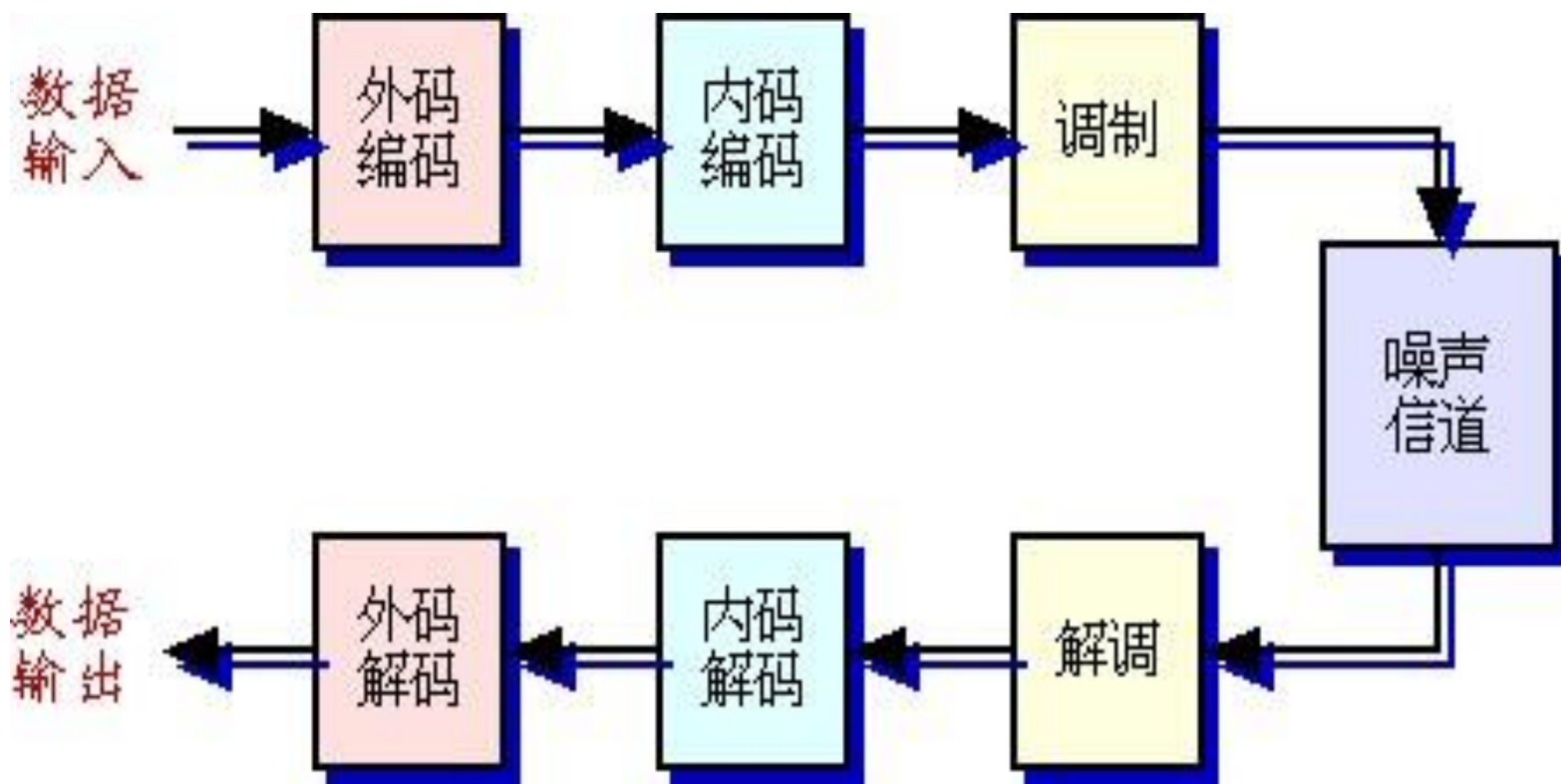
..... 13 10 7 4 17 14 11 8.....

- TCM技术
- 交织技术
- 级联码
- 差错检测和ARQ技术

- 简介

- 级联码是将两个独立的码级联起来构成一个性能更佳的码
- 一般是分组码与分组码级联、分组码与卷积码级联、或卷积码与卷积码级联
- 最常见的级联码是内码采用卷积码配合Viterbi译码，外码采用RS码
- 由于RS码具有很强的纠突发错误的能力，这种级联码特别适合于衰落信道

- 采用级联码的通信系统方框图



- 性能及应用

- 美国NASA在1977年发射的Voyager空间飞行器上采用级联码：

- 以 $(7, 1/2)$ 卷积码配合Viterbi译码为内码，以 $(255, 223)$ RS 码为外码
- 在误码率 P_b 为 10^{-6} 时所需的 E_b/N_0 为**2.53dB**，而只采用 $(7, 1/2)$ 卷积码则需约**4.5dB**

- 性能及应用(续)

- NASA在1989年发射的Galileo空间飞行器上再次采用级联码作为信道纠错码
 - 其内码为 $(15, 1/4)$ 卷积码配合Viterbi译码，外码为 $(255, 223)$ RS 码
 - 在误码率 P_b 为 10^{-6} 时所需的 E_b/N_0 为**0.91dB**，与只采用 $(15, 1/4)$ 卷积码相比，其编码增益约为**1.3dB**

级联码

表 不同级联码在误码率 P_b 为 10^{-6} 所需的 E_b/N_0

内 码 卷积码	外 码 RS 码	误码率 $P_b=10^{-6}$ 时 所需的 E_b/N_0 (dB)
$(7, 1/2)^a$	$(255, 223)$	2.53
$(15, 1/4)^b$	$(255, 223)$	0.91
$(13, 1/5)$	$(255, 229)$	0.91
$(14, 1/5)$	$(255, 231)$	0.79
$(14, 1/6)$	$(255, 233)$	0.63
$(14, 1/5)$	$(1023, 927)$	0.57
$(15, 1/5)$	$(1023, 959)$	0.50
$(14, 1/6)$	$(1023, 959)$	0.47
$(15, 1/6)$	$(1023, 959)$	0.42

^a 在 Voyager 空间飞行器上采用, ^b 在 Galileo 空间飞行器上采用。

级联码

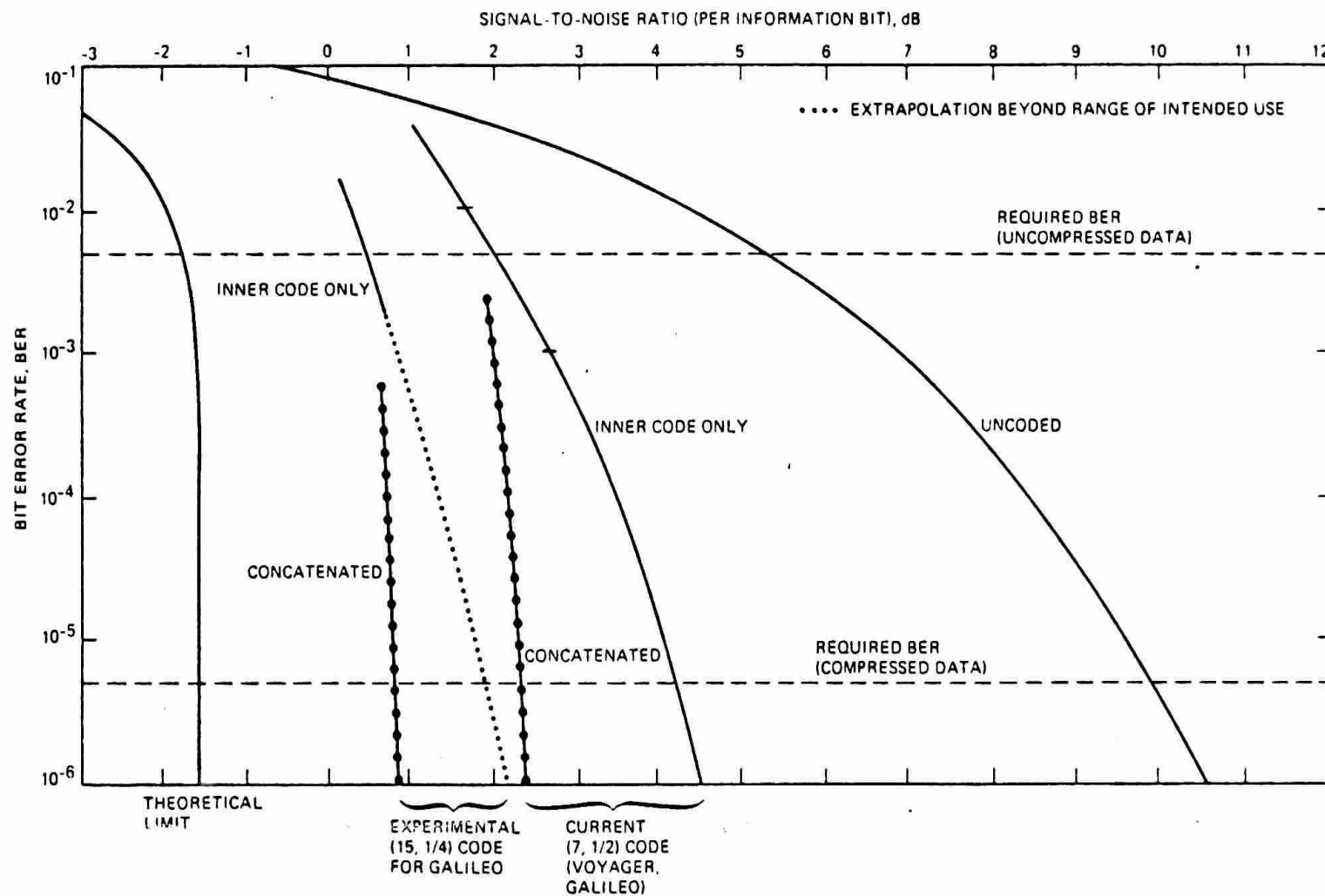


Fig. 3. Code performance.

- TCM技术
- 交织技术
- 级联码
- 差错检测和ARQ技术

差错检测和ARQ技术

- 前向差错控制技术不使用反向信道，直接利用信道编码技术纠正传输中的差错
- 差错检测和ARQ（Automatic Repeat reQuest）技术，利用信道编码技术发现传输中的差错，再利用反向信道通知发送方，请求发方重新发送
- 常用的差错检测方式：
 - 奇偶校验
 - CRC校验

- 在移动通信中应用的CRC校验：
 - 检错多项式：

$$g_1(x) = x^{12} + x^{11} + x^{10} + x^9 + x^8 + x^4 + x + 1$$

$$g_2(x) = x^8 + x^7 + x^4 + x^3 + x + 1$$

差错检测和ARQ技术

例：采用校验多项式 $g_2(x)$ ，

待发的nbit信息为：

$$c_{n-1}c_{n-2} \cdots c_1c_0$$

即
$$c(x) = c_{n-1}x^{n-1+8} + c_{n-2}x^{n-2+8} + \cdots + c_1x^9 + c_0x^8$$

信息序列的长度不受限制.

校验序列为 $c(x)$ 除 $g_2(x)$ 所得的余式 $r(x)$ 的系数：

$$r(x) = r_7x^7 + r_6x^6 + \cdots + r_1x + r_0$$

实际发送：

$$c_{n-1}c_{n-2} \cdots c_1c_0 \boxed{r_7r_6 \cdots r_1r_0}$$

例(续):

接收端用 $g_2(x)$ 去除接收码多项式:

如果整除, 则认为无错;

如果不能整除, 则认为有错, 请求发送方重发.

- HINOC2.0标准中CRC校验举例
R帧长为18比特，CRC校验码的生成多项式为：

$$g_2(x) = x^4 + x + 1$$

试分析其检错能力☺

THANKS