Curso de Especialização em Processamento de Sinais para Telecomunicações

Probabilidade e Teoria da Informação

Prof. Bartolomeu F. Uchôa Filho, Ph.D.

28 de agosto de 2005

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Revisão de Matemática

1.1 Análise Combinatória

No estudo que segue sobre probabilidade e sobre a análise de códigos corretores de erro, freqüentemente teremos a necessidade de enumerar ou contar certos elementos de um dado conjunto. Por exemplo, dado o conjunto de todos os resultados possíveis de um experimento, podemos querer saber o número de maneiras pelas quais um certo evento poderá ocorrer. Em certos casos, o procedimento de contagem ou enumeração pode ser bastante complicado. A Análise Combinatória é a parte da matemática que nos ajuda a contar os elementos de um conjunto nestas situações.

Exemplo 1 *Consideremos um lote de 100 peças contendo 20 peças defeituosas e 80 peças perfeitas. Escolhemos aleatoriamente 10 dentre as 100 peças do lote, sem reposição de qualquer peça escolhida antes que a seguinte seja escolhida. Estamos interessados em saber o número de possíveis maneiras pelas quais exatamente metade das 10 peças escolhidas é defeituosa.*

Depois de uma primeira análise, podemos dizer que a resposta para o problema de contagem acima é, se não difícil, no mínimo trabalhosa. Mas na verdade, como veremos, é humanamente impossível estimar a reposta para esse problema sem os conhecimentos desta seção. Devemos portanto recorrer a algumas técnicas sistemáticas de enumeração, as quais são apresentadas a seguir.

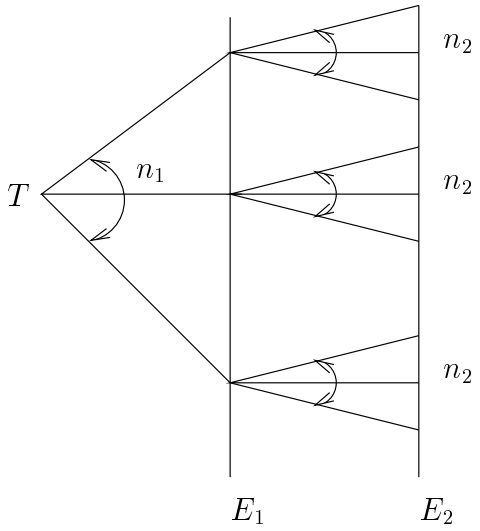


Figura 1.1: Árvore ilustrando a Regra da Multiplicação.

1.1.1 Regra da Multiplicação

Vamos supor que uma determinada tarefa, digamos T, possa ser realizada em duas etapas consecutivas, digamos E_1 e E_2. Vamos supor que a etapa E_1 possa ser executada de n_1 maneiras distintas, e que a etapa E_2 possa ser executada de n_2 maneiras distintas. Então, o número de maneiras distintas pelas quais a tarefa T pode ser realizada é

$$n_1 \cdot n_2.$$

Isto poderá ser visualizado na Figura 1.1.

Exemplo 2 *Considere uma palavra binária formada por 3 bits. Sabemos que cada bit pode ter o valor 0 ou 1. Assim, o número de triplas binárias é* $2.2.2 = 2^3 = 8$. *São elas: 000, 001, 010, 011, 100, 101, 110 e 111.*

1.1.2 Regra da Adição

Vamos supor que a etapa E_1 possa ser executada de n_1 maneiras e que a etapa E_2 possa ser executada de n_2 maneiras. Digamos que um determinado operário tenha que executar apenas uma das etapas. Ele deverá portanto ter a qualificação para executar uma etapa em

$$n_1 + n_2$$

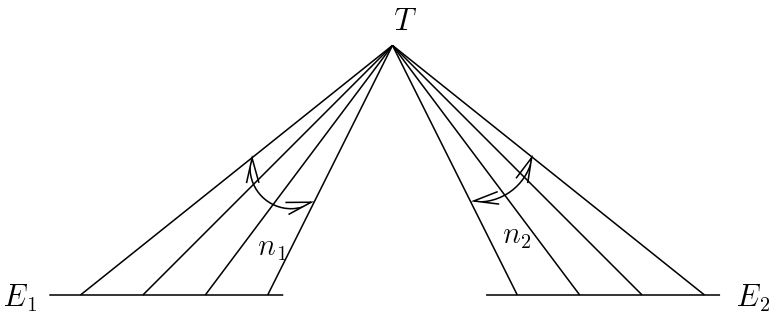


Figura 1.2: Árvore ilustrando a Regra da Adição.

maneiras. Isto poderá ser visualizado na Figura 1.2.

Exemplo 3 *Suponha que iremos planejar uma viagem e devemos escolher entre o transporte por ônibus ou por trem. Se existirem 3 rodovias e 2 ferrovias, então existirão* $3 + 2 = 5$ *caminhos disponíveis para a viagem.*

1.1.3 Permutações

Vamos supor que nós temos n objetos diferentes. Queremos saber o número ${}_nP_n$ de possíveis maneiras pelas quais podemos dispor esses objetos. Por exemplo, os números 1, 2 e 3 podem ser dispostos como 123, 132, 213, 231, 312 e 321. Portanto, neste caso a resposta é 6. Se n é um número grande, escrever todas as possíveis disposições dos n números é humanamente impossível. Estamos diante de um problema de *permutação.*

Para nos ajudar a demonstrar um procedimento sistemático desta enumeração, considere a caixa com n compartimentos mostrada na Figura 1.3. Permutar os n objetos é equivalente a colocá-los dentro dos compartimentos desta caixa, em alguma ordem particular. Colocando-se os objetos um a um, e da esquerda para a diretia, note que o primeiro compartimento poderá receber qualquer um dos n objetos. Já o segundo compartimento, poderá receber apenas $(n-1)$ dos n objetos, uma vez que um dos objetos já foi escolhido para ocupar o primeiro compartimento. Prosseguindo, restam $(n-2)$ possibilidades para preenchermos o terceiro compartimento, e assim por diante. Finalmente, o último compartimento só poderá ser prenchido de 1 maneira. Portanto, aplicando-se a Regra da Multiplicação, o número de permutações de n objetos

Figura 1.3: Caixa para auxiliar no cálculo de permutações e arranjos.

diferentes é dado por

$$ {}_nP_n = n(n-1)(n-2)\cdots 3\cdot 2\cdot 1 \triangleq n! $$

onde $n!$ é lido como o *fatorial* de n. Por definição, $0! = 1$. No exemplo anterior, temos $n = 3$. Assim, ${}_3P_3 = 3! = 3\cdot 2\cdot 1 = 6$, como já tínhamos encontrado.

1.1.4 Arranjos

Considere os mesmos n objetos distintos do caso anterior. Agora estamos interessados em escolher r objetos, onde $0 \leq r \leq n$, e permutar os r objetos escolhidos. Estamos pedindo o número ${}_nA_r$ de *arranjos* de r objetos escolhidos dentre n objetos distintos. Usaremos mais uma vez a caixa da Figura 1.3. Mas agora, preencheremos apenas até o r-ésimo compartimento, visto que estamos tratando do caso de um arranjo. O primeiro compartimento poderá ser preenchido de n maneiras distintas. O segundo, de $(n-1)$ maneiras distintas. E o r-ésimo compartimento poderá ser preenchido de $(n-(r-1))$ maneiras distintas. Portanto, o número total de arranjos é dado por:

$$\begin{aligned} {}_nA_r &= n(n-1)(n-2)\cdots(n-r+1) \\ &= \frac{n(n-1)(n-2)\cdots(n-r+1)(n-r)(n-r-1)\cdots 3\cdot 2\cdot 1}{(n-r)(n-r-1)\cdots 3\cdot 2\cdot 1} \\ &= \frac{n!}{(n-r)!}. \end{aligned}$$

Usando-se o exemplo anterior, quantos arranjos, ou seja, quantas maneiras há de escolher $r = 2$ objetos e permutar os $r = 2$ objetos escolhidos, dentre $n = 3$ objetos distintos? Temos, para os números 1, 2 e 3, os seguintes arranjos: 12, 13, 23, 21, 31 e 32. Ou seja, $\frac{n!}{(n-r)!} = \frac{3!}{1!} = \frac{6}{1} = 6$ arranjos.

1.1.5 Combinações

Considere os mesmos n objetos distintos dos casos anteriores. Agora estamos interessados no número C_r^n de maneiras de escolher r objetos, onde $0 \leq r \leq n$, dentre os n objetos, porém sem nos preocuparmos com a ordem. Por exemplo, dentre os números 1, 2, 3 e 4, quantas maneiras há de escolher $r = 2$ números distintos, sem considerar a ordem? Poderemos ter: 12, 13, 14, 23, 24 e 34. Note, por exemplo, que não contamos a combinação 21, pois os dois objetos (1 e 2, no caso) são os mesmos da combinação 12 e apenas a ordem é diferente. Para obtermos C_r^n no caso geral, podemos usar a fórmula para o número total de arranjos. Note que, quando consideramos o número de arranjos, ${}_nA_r$, contamos todas as possíveis ordens de cada grupo de r objetos. Portanto, para obtermos C_r^n basta dividirmos ${}_nA_r$ pelo número de permutações de r objetos distintos. Assim,

$$C_r^n = \frac{n!}{r!(n-r)!} \triangleq \binom{n}{r}$$

Os números $\binom{n}{r}$ são chamados de *coeficientes binomiais*.

Devemos agora tentar resolver o problema do Exemplo 1 com os conhecimentos adquiridos nesta seção. Escolher 5 peças defeituosas dentre 20 delas, sem nos preocuparmos com a ordem, pode ser feito de $\binom{20}{5}$ maneiras. Para cada uma dessas maneiras, temos $\binom{80}{5}$ possibilidades para a escolha de 5 peças perfeitas dentre 80 peças perfeitas. Assim, usando a Regra da Multiplicação, teremos $\binom{20}{5} \cdot \binom{80}{5} \approx 3,72 \times 10^{11}$ maneiras possíveis pelas quais exatamente metade das 10 peças escolhidas é defeituosa, resolvendo o problema do Exemplo 1.

OBS.: Note que há $\binom{100}{10}$ maneiras de escolhermos 10 peças quaisquer dentre 100 peças, sem reposição e sem nos preocupar com a ordem. Logo, a probabilidade de escolhermos 10 peças dentre 100 com exatamente 5 peças defeituosas é:

$$\frac{\binom{20}{5}\binom{80}{5}}{\binom{100}{10}} = 0,021 \text{ ou } 2,1\%.$$

Esta é conhecida como *probabilidade geométrica*.

PROBLEMAS:

1. Dentre 8 pessoas, quantas comissões de 3 membros podem ser escolhidas?

2. Com 8 bandeiras distintas, quantos sinais com 3 bandeiras se pode obter?

3. De um grupo de 8 profissionais formado de 5 graduados e 3 de nível médio, quantas equipes de 3 profissionais podem ser constituídas, incluindo exatamente 2 graduados?

4. Quantos *bytes* de 8 bits há com exatamente dois 1's?

5. Quantos *bytes* de 8 bits há com um número par de 1's?

6. Quantos *bytes* de 8 bits há?

7. Quantos são os subconjuntos de um conjunto constituído de 8 elementos (contados o conjunto vazio e o próprio conjunto)?

8. Obtenha uma expressão geral para $(a+b)^n$, onde a e b são números reais quaisquer e n é um inteiro não negativo.

1.2 Logaritmos

Nesta seção vamos rever algumas propriedades de logaritmos. Elas se mostrarão necessárias nas manipulações algébricas e nas provas dos resultados de Teoria da Informação.

Sejam A e B números reais positivos quaisquer, e b e c números inteiros quaisquer. Então, temos as seguintes propriedades:

1. $\log_b 1 = 0$

2. $\log_b A^n = n \log_b A$

3. $\log_b AB = \log_b A + \log_b B$

4. $$\log_b \left(\frac{A}{B}\right) = \log_b A - \log_b B \Rightarrow \log_b \left(\frac{1}{B}\right) = -\log_b B$$

5. $$\log_b A = \frac{\log_c A}{\log_c b}$$

OBS.: Note que a propriedade 5 é útil para obtermos logaritmos em uma base qualquer, quando na nossa calculadora só temos as bases 10 e neperiana.

PROBLEMAS:

1. Calcule $\log_3 27$.
2. Calcule $\log_{10} 1.000.000.000$.
3. Dado que $y = e^{-\ln(1/x)}$, mostre que $x = y$.

Capítulo 2

Probabilidade I: Variáveis Aleatórias Discretas

Quando realizamos um experimento e o resultado deste experimento não pode ser previsto com certeza, dizemos que se trata de um *experimento aleatório*. A ausência de um modelo determinístico para certos fenômenos, os chamados fenômenos aleatórios, pode ser uma consequência da nossa parcial ou total ignorância a respeito dos fenômenos, ou pela dificuldade técnica ou matemática de se obter esse modelo, ou por qualquer outra razão, fora do nosso controle, que nos torne incapazes de prever com certeza o comportamento destes fenômenos. Nestes casos, consideramos um modelo probabilístico. Tais modelos, apesar de serem não-determinísticos, fornecem informações que podem ser consideradas suficientes para certos propósitos. Modelos probabilísticos são amplamente utilizados nas mais diversas áreas, inclusive, é claro, Comunicações.

2.1 Espaço Amostral

Inicialmente vamos introduzir o conceito de espaço amostral. Denotado por Ω, o *espaço amostral* é o conjunto de todos os possíveis resultados do experimento aleatório em questão.

Exemplo 4 *Se considerarmos o lançamento de um dado com seis faces, o espaço amostral dos possíveis resultados será* $\Omega = \{1, 2, 3, 4, 5, 6\}$. *No caso do lançamento de*

uma moeda, teremos que $\Omega = \{cara, coroa\}$.

Outro conceito importante no nosso estudo é o de evento. Um *evento* é qualquer subconjunto do espaço amostral.

Exemplo 5 *Tomando como exemplo o caso do dado, alguns dos possíveis eventos são:*

- $A = \{4, 5, 6\} \subset \Omega$
- $B = \{1\} \subset \Omega$
- $\phi \subset \Omega$ *(*evento impossível*)*
- Ω *(*evento certo*)*

Note que Ω e ϕ também são considerados eventos. Em particular, estes são os chamados *eventos triviais*. Os eventos que contêm um único resultado de um experimento, como no caso do evento B acima, são chamados de *eventos elementares* ou *atômicos*.

É importante distinguirmos entre um elemento de um conjunto com um único elemento e o próprio conjunto. Neste sentido, dizemos que $4 \in A$, mas $\{4\} \subset A$.

2.2 Medida de Probabilidade, P

Uma medida de probabilidade P associa a cada evento um número real no intervalo [0,1].

Exemplo 6 *Consideremos o caso do dado de seis faces e os eventos A e B do Exemplo 5. Supondo que o dado seja equilibrado, teremos:*

$$P(A) = \frac{3}{6} = \frac{1}{2}\,,$$

$$P(B) = \frac{1}{6}, \quad P(\phi) = 0 \quad e \quad P(\Omega) = 1$$

A medida de probabilidade P deve satisfazer as seguintes condições:

1. $P(A) \geq 0$

2. $P(\Omega) = 1$

3. Sejam dois eventos A e B tais que $A \cap B = \phi$. Então,

$$P(A \cup B) = P(A) + P(B)$$

Assim, para o Exemplo 6, teremos:

$$P(A \cup B) = P(\{1, 4, 5, 6\}) = P(A) + P(B) = \frac{1}{2} + \frac{1}{6} = \frac{4}{6}$$

As condições 1, 2 e 3 acima são conhecidas como os *axiomas* da probabilidade P. A definição axiomática de probabilidade é importante para a demonstração de uma série de propriedades. Mas, uma definição mais intuitiva de probabilidade é a definição por meio de freqüência relativa. Esta será considerada mais adiante.

2.3 Probabilidade de Eventos Elementares

Para $\Omega = \{s_1, s_2, \ldots, s_n\}$, definimos a probabilidade de um evento elementar por:

$$p_i = P(\{s_i\}) \quad , \quad i = 1, 2, \ldots, n$$

2.4 Variável Aleatória Discreta

Uma variável aleatória (v.a.) discreta é um mapeamento (função) do espaço amostral em um conjunto específico finito ou infinito, porém numerável (ou contável). No caso de o conjunto ser infinito e não-numerável, a v.a. será contínua. Estudaremos v.a.'s contínuas no Capítulo 6. O processo de geração de uma variável aleatória é mostrado na Figura 2.1.

Exemplo 7 *Para* $\Omega = \{cara, coroa\}$, *podemos ter*

$$\begin{aligned} X(cara) &= 0 \qquad X(\Omega) = \{0, 1\} \quad \textit{(novo espaço amostral)} \\ X(coroa) &= 1 \end{aligned}$$

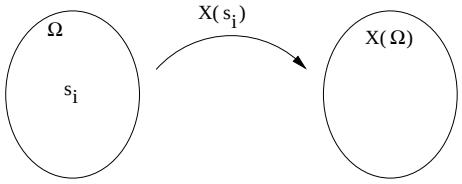


Figura 2.1: Geração de uma variável aleatória.

Note que o conceito de v.a. é importante, pois com ele podemos falar de um experimento universal. Não importa se estamos falando do lançamento de uma moeda, da seleção de uma peça perfeita ou defeituosa, de ganhar ou perder, ou de qualquer outro experimento com dois possíveis resultados. Pelo conceito de v.a. binária, todos esses experimentos são essencialmente o mesmo, com espaço amostral $X(\Omega) = \{0, 1\}$.

2.5 Distribuição de Probabilidade de uma v.a. Discreta

Uma distribuição de probabilidades de uma v.a. discreta X é um mapeamento, denotado por P_X, de $X(\Omega)$ para o intervalo [0:1].

$$\begin{aligned} P_X(x) &= P(X = x) \\ &= \text{probabilidade de } X = x \\ &= \text{probabilidade do evento } \{s : X(s) = x\} \end{aligned}$$

Exemplo 8 *Para* $\Omega = \{sol, chuva, nublado\}$*, podemos ter:*

$$\begin{aligned} X(sol) &= 0 \\ X(chuva) &= 1 \qquad P(X = 1) = \textit{probabilidade de chuva} \\ X(nublado) &= 2 \end{aligned}$$

Como P_X é uma probabilidade relacionada aos eventos atômicos *sol*, *chuva* e *nublado*, temos:

$$P_X(x) \geq 0 \quad \text{e} \sum_{x \in X(\Omega)} P_X(x) = 1$$

2.6 Probabilidades Conjunta, Condicionada e Total e Independência Estatística

Para entendermos os conceitos de probabilidades *conjunta*, *condicionada* e *total*, bem como o conceito de independência estatística, vejamos o seguinte exemplo:

Exemplo 9 *Vamos considerar um experimento para determinar o clima na cidade de Florianópolis. Considere os três eventos A, B e C:*

$$\begin{aligned} A: &\ \textit{num dado dia, a temperatura é} \ \geq \ 10^o C \\ B: &\ \textit{num dado dia, chove} \ \geq \ 5\ mm \\ C: &\ \textit{num dado dia, a temperatura é} \ \geq \ 10^o C \ \textit{e chove} \geq 5\ mm \end{aligned}$$

Note que $C = A \cap B$. A *interseção* é também representada por $C \triangleq AB$. A *probabilidade conjunta* é definida como:

$$P(C) = P(AB)$$

que é a probabilidade de os eventos A e B ocorrerem simultaneamente. Podemos facilmente estender este conceito para 3 ou mais eventos. Por exemplo, se E, F e G são eventos quaisquer, a probabilidade conjunta destes três eventos será escrita como $P(EFG)$.

Voltando ao Exemplo 9, seja n_i o número de dias em que ocorreu o evento i, onde $i = A, B$ ou C. Vamos supor que depois de observarmos o clima na cidade de Florianópolis por um período de $n = 1000$ dias, encontramos:

$$n_A = 811 \qquad n_B = 306 \qquad n_{AB} = 283$$

como pode ser observado no diagrama de Venn da Figura 2.2. Assim,

$$P(A) \cong \frac{n_A}{n} = \frac{811}{1000} = 0,811$$

$$P(B) \cong \frac{n_B}{n} = 0,306$$

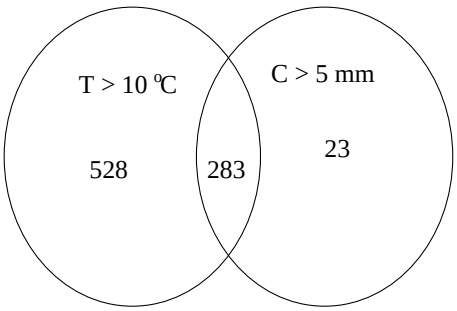


Figura 2.2: Diagrama de Venn para o problema do Exemplo 9.

$$P(AB) \cong \frac{n_{AB}}{n} = 0,283$$

Note que estamos tendo o cuidado de escrever que a probabilidade é aproximadamente ($\cong$) dada pela freqüência relativa. Segundo a definição de probabilidade pela freqüência relativa, alternativa à definição axiomática, a probabilidade é obtida se o número n de experimentos for infinitamente grande, o que não é o caso no exemplo acima.

Consideremos agora a relação $\frac{n_{AB}}{n_A}$. O que ela representa? Note que estamos computando a fração do número de dias em que a temperatura é maior que 10^oC na qual tenha chovido pelo menos 5 mm. Esta fração portanto representa aproximadamente a probabilidade de chover pelo menos 5 mm dado que a temperatura em um certo dia é maior ou igual a 10^oC. De modo mais geral, estamos considerando a probabilidade de ocorrer um evento B dado que um evento A tenha ocorrido. Esta é chamada de *probabilidade condicionada*. Sucintamente, dizemos que esta é a probabilidade de B dado A, e a denotamos por $P(B|A)$. Assim,

$$P(B|A) \cong \frac{n_{AB}}{n_A} = \frac{\frac{n_{AB}}{n}}{\frac{n_A}{n}} \triangleq \frac{P(AB)}{P(A)}$$

A expressão acima, que traz uma relação importante entre as probabilidades condicionada e conjunta, é conhecida como a *Regra de Bayes*, que é formalmente apresentada a seguir:

$$P(B|A) \triangleq \frac{P(AB)}{P(A)}, \text{ para } P(A) > 0$$

ou

$$P(A|B) \triangleq \frac{P(AB)}{P(B)}, \text{ para } P(B) > 0$$

Ambas as fórmulas acima são equivalentes. A razão para escrevermos aqui a segunda fórmula é que ela simplifica o entendimento de certas demonstrações matemáticas

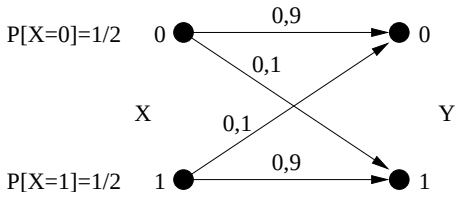


Figura 2.3: Canal binário simétrico com probabilidade de transição 0,1.

que encontraremos mais adiante. Para demonstrar os conceitos acima, vejamos um exemplo.

Exemplo 10 *Vamos considerar o* Canal Binário Simétrico *(ou BSC, do inglês "binary symmetric channel") descrito na Figura 2.3. Esse canal é definido pelas seguintes probabilidades condicionadas:*

$$P(Y=1|X=1) = P(Y=0|X=0) = 0,9 \quad \textit{(probabilidade de acerto)}$$

e

$$P(Y=1|X=0) = P(Y=0|X=1) = 0,1 \quad \textit{(probabilidade de erro)}$$

e pelas probabilidades a priori*:*

$$P(X=0) = P(X=1) = \frac{1}{2}.$$

Calcule a probabilidade conjunta: $P(X=0, Y=0)$.

Solução: Pela regra de Bayes, podemos escrever a probabilidade conjunta de duas maneiras:

$$P(X=0,\ Y=0) = P(X=0|Y=0)\ \ P(Y=0)$$

ou

$$P(X=0,\ Y=0) = P(Y=0|X=0)\ \ P(X=0)$$

Pelos dados do problema, devemos escolher a segunda maneira, o que fornece:

$$P(X=0,\ Y=0) = 0,9 \times \frac{1}{2} = 0,45.$$

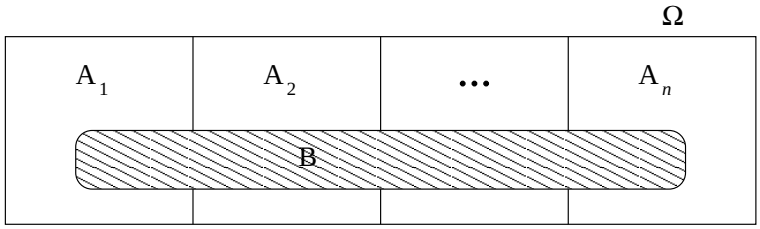


Figura 2.4: Teorema da probabilidade total.

2.7 Teorema da Probabilidade Total

Sejam $A_1, A_2, \ldots, A_n$, n eventos mutuamente excludentes e exaustivos, ou seja:

$$\bigcap_{i=1}^{n} A_i = \phi \quad \text{e} \quad \bigcup_{i=1}^{n} A_i = \Omega,$$

respectivamente. Esta situação é ilustrada na Figura 2.4. Supondo que $P(A_i) \neq 0, \forall i$, temos que:

$$P(B) = P(B|A_1)P(A_1) + P(B|A_2)P(A_2) + \ldots + P(B|A_n)P(A_n).$$

Este resultado é conhecido como o *Teorema da Probabilidade Total*.

Exemplo 11 *De volta ao Exemplo 10, calcule* $P(Y = 0)$.

Solução*:*

$$P(Y = 0) = P(Y = 0|X = 0)P(X = 0) + P(Y = 0|X = 1)P(X = 1)$$
$$= 0,1 \times \frac{1}{2} + 0,9 \times \frac{1}{2} = \frac{1}{2}$$

Exemplo 12 *Calcule* $P(X = 0|Y = 0)$.

Solução*:*

$$P(X = 0|Y = 0) = \frac{P(X = 0, Y = 0)}{P(Y = 0)} = \frac{0,45}{\frac{1}{2}} = 0,9$$

2.8 Independência Estatística:

Dois eventos $A \subset \Omega$ e $B \subset \Omega$ com $P(A) > 0$ e $P(B) > 0$ são *estatisticamente independentes* se e somente se

$$P(AB) = P(A)P(B)$$

Note que, como

$$P(AB) = P(B|A)P(A)$$

temos que se os eventos A e B são estatisticamente independentes, então

$$P(B|A) = P(B)$$

Ou seja, o fato de o evento A ter ocorrido não muda a probabilidade de o evento B ocorrer. Daí se dizer que os eventos A e B são estatisticamente independentes. Similarmente, como

$$P(AB) = P(A|B)P(B)$$

também temos que

$$P(A|B) = P(A)$$

e da mesma maneira o conhecimento de B não afeta as chances de A ocorrer. Vejamos um exemplo.

Exemplo 13 *Considere o experimento caracterizado pelo lançamento simultâneo de uma moeda eqüilibrada e de um dado de seis faces também eqüilibrado. Ou seja, o espaço amostral é:*

$$\Omega = \{(cara, 1); (cara, 2), \ldots, (cara, 6); (coroa, 1), \ldots, (coroa, 6)\}$$

Intuitivamente, podemos dizer que o resultado do experimento com a moeda é independente daquele realizado com o dado. Assim, a probabilidade de obtermos uma cara *na moeda e simultaneamente o número 6 no dado pode ser obtida por:*

$$P(coroa, 6) = P(coroa) \times P(6) = \frac{1}{2} \times \frac{1}{6} = \frac{1}{12}$$

2.9 Teorema de Bayes

Sejam $A_i,\ i = 1, 2, \ldots, n$, eventos exaustivos e disjuntos (ou seja, $\cup_i A_i = \Omega$ e $A_i \cap A_j = \phi,\ i \neq j$) com $P(A_i) > 0,\ \forall i$. Seja B um evento tal que $P(B) > 0$. Então:

$$P(A_j|B) = \frac{P(B|A_j) \times P(A_j)}{\sum_{i=1}^{n} P(B|A_i)P(A_i)}$$

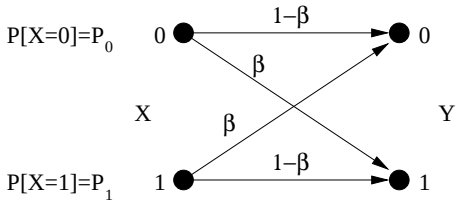


Figura 2.5: Canal binário simétrico com probabilidade de transição β.

Exemplo 14 *Considere o Canal Binário Simétrico mostrado na Figura 2.5. Diferentemente da situação do Exemplo 10, aqui os símbolos de entrada são não-eqüiprováveis. Ou seja,*

$$P(X=0) = P_0 \quad P(X=1) = 1 - P_0 \stackrel{\Delta}{=} P_1$$

onde possivelmente $P_0 \neq 1/2$. *Calcule* $P(X=1|Y=1)$.

Solução*:*

$$\begin{aligned} P(X=1|Y=1) &= \frac{P(X=1,Y=1)}{P(Y=1)} \\ &= \frac{P(Y=1|X=1)\,P(X=1)}{P(Y=1|X=0)\,P(X=0) + P(Y=1|X=1)\,P(X=1)} \\ &= \frac{P_1(1-\beta)}{P_0\beta + P_1(1-\beta)} \end{aligned}$$

Se $P_0 = P_1 = \frac{1}{2}$, *teremos:*

$$P(X=1|Y=1) = 1 - \beta$$

Exemplo 15 *Consideremos agora um experimento com um radar e definamos os seguintes eventos:*

$$\begin{aligned} A &= \textit{radar detecta um alvo} \\ B &= \textit{o alvo de fato existe} \\ \overline{A} &= \textit{o radar não detecta nada} \\ \overline{B} &= \textit{não há alvo} \end{aligned}$$

Conhecemos os seguintes dados do problema: $P(A|B) = 0,95$, $P(\overline{A}|\overline{B}) = 0,95$ *e* $P(B) = 0,005$. *Calcule* $P(B|A)$. *O radar é eficiente?*

Solução: Temos que

$$(\overline{B}|A) = 1 - P(B|A) = \text{probabilidade de alarme falso}$$

$$P(\overline{A}|B) = 1 - P(A|B) = 0,05 = \text{ probabilidade de falhar na detecção}$$

Assim,

$$P(B|A) = \frac{P(A|B)\,P(B)}{P(A|B)\,P(B) + P(A|\overline{B})\,P(\overline{B})}$$
$$= \frac{0,95 \times 0,005}{0,95 \times 0,005 + [1 - P(\overline{A}|\overline{B})][1 - P(B)]}$$
$$= \frac{0,95 \times 0,005}{0,95 \times 0,005 + 0,05 \times 0,995} = 0,087 = 8,7\%$$

2.10 Esperança de F(X):

Seja F uma função real cujo domínio inclui $X(\Omega)$. A esperança de F(X) é o número real

$$E\{F(X)\} = \overline{F(X)} = \sum_{x \in X} P_X(x)\,F(x)$$

2.10.1 Casos Particulares:

1. *Média*:

$$F(X) = X \Rightarrow E\{X\} \stackrel{\Delta}{=} \overline{X} = \sum_{x \in X} x\,P_X(x)$$

2. *Variância*:

$$F(X) = (X - \overline{X})^2 \Rightarrow E\{(X - \overline{X})^2\} = \sum_{x \in X} (x - \overline{X})^2\,P_X(x) = \mathrm{Var}(X) \stackrel{\Delta}{=} \sigma_X^2$$

Analogamente, podemos ter a definição de esperança para duas variáveis aleatórias, digamos X e Y, da seguinte forma:

$$E\{F(X,Y)\} = \sum_{x \in X} \sum_{y \in Y} P_{X,Y}(x,y)\,F(x,y)$$

Exemplo 16 *Para $F(X,Y) = XY$, temos a correlação cruzada. Veremos este conceito com mais detalhes no estudo de processos estocáticos no Capítulo 6.*

2.11 Esperança Condicionada

Podemos ter esperança de $F(X)$ condicionada à ocorrência do evento $Y = y$. Escrevemos então:

$$E\{F(X)|Y = y\} = \sum_x F(x)P_{X|Y}(x|y)$$

Sempre que necessário, usaremos $P_{X|Y}(x|y) = \frac{P_{XY}(x,y)}{P_Y(y)}$.

Note que, se $E\{F(X)|Y = y\}$ é conhecida para todo $y \in Y$, então devemos ter

$$\begin{aligned}
E\{F(X)\} &= \sum_{y\in Y} E\{F(X)|Y = y\}\ P_Y(y) \\
&= \sum_{y\in Y}\sum_{x\in X} F(x)P_{X|Y}(x|y)P_Y(y) \\
&= \sum_{x\in X}\sum_{y\in Y} F(x)P_{XY}(x,y) \\
&= \sum_{x\in X} F(x)\sum_{y\in Y} P_{XY}(x,y) \\
&= \sum_{x\in X} F(x)P_X(x) \\
&= E\{F(X)\}
\end{aligned}$$

Note que na penúltima igualdade com somatória fizemos uso da relação

$$\sum_{y\in Y} P_{XY}(x,y) = P_X(x)$$

Quando obtemos $P_X(x)$ desta maneira, chamamos $P_X(x)$ de *Distribuição de Probabilidade Marginal.*

Exemplo 17 *Vamos supor que uma empresa de comunicação de dados tenha a opção de usar alguns tipos de canais, listados na Tabela 2.1. Vamos admitir que a escolha do canal é baseada na disponibilidade, que é um fenômeno aleatório. Vamos supor que* $P(Y = i) = 1/4$, $i = 1, 2, 3, 4$, *onde* $P(Y = i)$ *é a probabilidade de o canal* i *ser escolhido. Seja* X *o atraso médio da mensagem em milisegundos, que é diferente para*

Tabela 2.1: Dados para o problema do Exemplo 17

Y	Tipo de Canal
1	satélite
2	cabo coaxial
3	enlace de microondas
4	fibra óptica

cada canal, segundo as seguintes esperanças condicionadas:

$$\begin{aligned} E\{X|Y=1\} &= 500\ ms \\ E\{X|Y=2\} &= 300\ ms \\ E\{X|Y=3\} &= 200\ ms \\ E\{X|Y=4\} &= 100\ ms \end{aligned}$$

Calcule o atraso médio $E\{X\}$.

Temos que

$$E\{X\} = \sum_{y=1}^{4} E\{X|Y=y\}P(y)$$

$$= 500 \times \frac{1}{4} + 300 \times \frac{1}{4} + 200 \times \frac{1}{4} + 100 \times \frac{1}{4} = 275 \text{ ms}$$

2.12 Exemplos de Distribuição de Variáveis Aleatórias Discretas

A seguir, apresentamos algumas das distribuições de probabilidades encontradas no nosso estudo, com as respectivas média e variância.

1. Uniforme: $\Omega = \{1, 2, \ldots, n\}$, com $P(X = i) = 1/n$, $\forall i \in \Omega$. A média e a variância são dadas por:

$$E\{X\} = \frac{1}{n}\sum_{i=1}^{n} i = \frac{1}{n}\,(1+n)\,\frac{n}{2} = \frac{n+1}{2}$$

$$\mathrm{Var}(X) = \frac{1}{n}\sum_{i=1}^{n}(i - \frac{n+1}{2})^2 = \frac{n^2-1}{12}$$

2. Bernoulli: $\Omega = \{0, 1\}$, com $P(X = 0) = 1 - p$ e $P(X = 1) = p$. A média e a variância são dadas por:

$$E\{X\} = 0 \times (1-p) + 1 \times p = p$$

$$\begin{aligned} \mathrm{Var}(X) &= (0-p)^2(1-p) + (1-p)^2 p = p^2 - p^3 + p - 2p^2 + p^3 = p - p^2 \\ &= p(1-p) \end{aligned}$$

3. $\Omega = \{1, 2, \ldots, n\}$, com $P(X = i) = 1$ para algum i, e $P(X = j) = 0$ para $j \neq i$. A média e a variância são dadas por:

$$E\{X\} = i \times 1 = i$$

$$\mathrm{Var}(X) = (i-i)^2 \times 1 = 0$$

PROBLEMAS:

1. Um experimento aleatório consiste em sortear uma bola de uma urna que contém quatro bolas vermelhas numeradas com 1, 2, 3 e 4, e três bolas pretas numeradas com 1, 2 e 3. Determine com precisão que resultados do experimento acima estão contidos nos seguintes eventos:

   (a) $E_1 =$ O número de bolas é par.

   (b) $E_2 =$ A cor da bola é vermelha e seu número é maior que 1.

   (c) $E_3 =$ O número da bola é menor que 3.

   (d) $E_4 = E_1 \cup E_3$.

   (e) $E_5 = E_1 \cup (E_2 \cap E_3)$.

2. Se todas as bolas no problema anterior são sorteadas da urna com a mesma probabilidade, encontre as probabilidades de E_i, $1 \leq i \leq 5$.

3. Em uma certa cidade, três marcas de carro A, B e C têm 20%, 30% e 50% da preferência, respectivamente. A probabilidade de que um carro necessite de manutenção durante seu primeiro ano desde a compra para as três marcas são 5%, 10% e 15%, respectivamente.

   (a) Qual a probabilidade de um carro nessa cidade necessitar de manutenção durante seu primeiro ano desde a compra?

   (b) Se um carro nesta cidade necessitar de manutenção durante seu primeiro ano desde a compra, qual a probabilidade de esse carro ser da marca A?

4. Em que condições dois eventos A e B disjuntos podem ser independentes?

5. Um fonte de informação produz 0 e 1 com probabilidades 0,3 e 0,7, respectivamente. A saída da fonte é transmitida por um canal cuja probabilidade de erro (ou seja, a probabilidade de 1 se transformar em 0, ou vice-versa) é 0,2.

   (a) Qual a probabilidade de na saída do canal ser observado um 1?

   (b) Qual a probabilidade de a saída da fonte ter sido 1 dado que a saída observada do canal foi 1?

6. Uma moeda é lançada três vezes e a variável aleatória discreta X denota o número total de *caras* que aparecem. A probabilidade de em um lançamento da moeda sair uma *cara* é denotada por p.

   (a) Quantos valores a variável aleatória X pode ter?

   (b) Qual a distribuição de probabilidades de X?

   (c) Qual a probabilidade de X ser maior que 1?

   OBS.: Supor que os resultados dos três lançamentos da moeda sejam estatisticamente independentes.

7. A moeda A tem probabilidade de *cara* igual a 0,25 e de *coroa* 0,75. A moeda B é equilibrada. Cada moeda é lançada 4 vezes. Considere a variável aleatória discreta X que denota o número de *caras* resultantes da moeda A e Y que denota o número de *caras* resultantes da moeda B.

   (a) Qual a probabilidade de $X = Y = 2$?

   (b) Qual a probabilidade de $X = Y$?

   (c) Qual a probabilidade de $X > Y$?

   (d) Qual a probabilidade de $X + Y \leq 5$?

8. Considere a variável aleatória $X = \{1, 2, 3, 4\}$, com probabilidades $P_X(X = 1) = 0,4$, $P_X(X = 2) = 0,25$, $P_X(X = 3) = 0,2$ e $P_X(X = 4) = 0,15$. Calcule a esperança de $F(X)$ para as seguintes funções:

   (a) $F(X) = 1$.

   (b) $F(X) = X$.

   (c) $F(X) = X^2$.

   (d) $F(X) = (X - \overline{X})^2$, onde $\overline{X}$ é a esperança obtida no item b.

9. Considerando o problema 5, calcule a esperança condicionada $E\{F(X)|Y = 1\}$, onde $F(X) = X$.

Capítulo 3

Teoria da Informação

3.1 Introdução

O matemático americano Claude E. Shannon, em 1948, publicou uma série de resultados que se tornaram conhecidos como a "Teoria Matemática das Comunicações", hoje conhecida como Teoria da Informação. Shannon definiu uma medida quantitativa para informação e estabeleceu os limites para uma comunicação confiável, quando demonstrou matematicamente que, se a taxa de transmissão for mantida abaixo da capacidade de canal, medida esta por ele próprio definida e determinada para alguns tipos de canais, uma comunicação confiável, ou seja, com uma probabilidade de erro tão pequena quanto se queira, poderá ser alcançada se forem apropriadamente escolhidos codificadores e decodificadores de mensagens. Algumas formas simples de codificação e decodificação de mensagens foram propostas pelo próprio Shannon, porém muito mais como instrumentos para auxiliá-lo nas provas dos seus teoremas do que como maneiras eficientes de se obter tamanho grau de confiabilidade na comunicação. Em suma, Shannon demonstrou a existência de codificadores e decodificadores eficientes, porém não nos ensinou como obtê-los.

A Teoria de Códigos Corretores de Erro surgiu no começo da década de 50. Buscava-se codificadores e decodificadores de mensagens eficientes, cujo desempenho se aproximasse dos limites estabelecidos por Shannon, e cuja complexidade fosse baixa o suficiente para que pudessem ser implementados na prática. O trabalho inicial para

a obtenção das primeiras classes de bons códigos corretores de erro foi árduo, pois exigia um profundo conhecimento de matemática (álgebra abstrata, teoria de probabilidade, etc.), sendo desenvolvido por um grupo restrito composto basicamente por matemáticos, embora a importância prática daquele tema já fosse reconhecida pelos engenheiros de comunicações da época. Décadas se passaram e hoje se tem conhecimento de várias classes de bons códigos corretores de erro, que podem ser perfeitamente entendidas por engenheiros e cientistas de computação, graças ao esforço de vários pesquisadores, que souberam apresentar esse material com um mínimo rigor matemático.

Os códigos corretores de erro estão presentes nas mais diversas situações. Por exemplo, códigos corretores de erro foram utilizados na transmissão dos sinais de vídeo mostrando os primeiros passos de um ser humano no solo lunar. Também garantem a qualidade das ligações telefônicas interurbanas e internacionais, bem como da telefonia celular. Hoje os vendedores de assinatura de TV digital via satélite anunciam os tão desejáveis "som com qualidade de CD" e imagens perfeitas graças à presença de códigos corretores de erro naqueles sistemas. MODEMS modernos que permitem rápidas conexões com a Internet usam códigos corretores de erro, bem como todos os sistemas de armazenagem de dados em meios ópticos ou magnéticos, como discos rígidos, CD's, DVD's, etc.

Neste capítulo, estudaremos os conceitos fundamentais de Teoria da Informação. Uma introdução aos Códigos Corretores de Erro é o assunto do Capítulo 4.

3.2 Sistema de Comunicação Digital

Antes de iniciarmos o nosso estudo sobre Teoria da Informação, devemos conhecer alguns componentes básicos que fazem parte de um sistema de comunicação digital. É importante salientar que outros componentes importantes existem, mas por não serem objetos de nosso estudo, estes não serão aqui considerados. Considere o diagrama de blocos de um sistema de comunicação digital da Figura 3.1. Na extremidade superior esquerda, temos uma *fonte discreta*. Uma fonte discreta é qualquer dispositivo que emita símbolos a partir de um certo alfabeto finito, formando uma seqüência. Por exemplo, se o alfabeto for binário (ex., $\{0, 1\}$), teremos que a fonte discreta emitirá

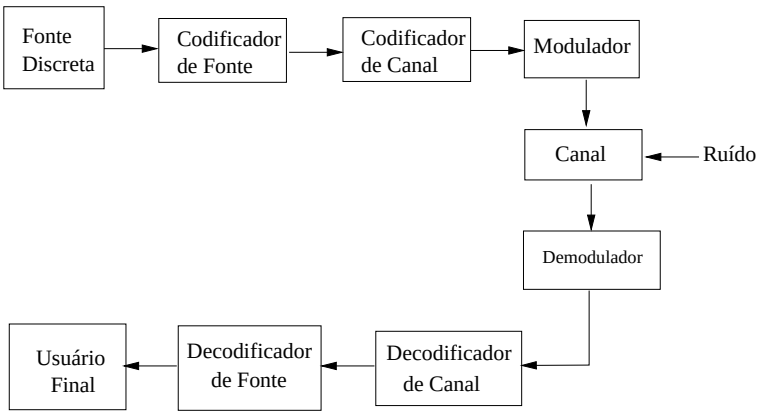


Figura 3.1: Sistema de comunicação digital.

uma seqüência binária, de zeros e uns. Se há a possibilidade de a fonte emitir mais de uma seqüência, diremos que a fonte produz informação. Por essa razão, muitas vezes chamaremos este componente do sistema de comunicação digital de *fonte de informação*. Os conceitos precisos de informação e de produção de informação serão apresentados mais adiante neste capítulo. O próximo componente na Figura 3.1 é o *codificador de fonte*, o qual será brevemente descrito na Seção 3.2.1. Em seguida, temos o *codificador de canal*, sobre o qual falaremos superficialmente na Seção 3.2.2 e com uma certa profundidade no Capítulo 4. O *modulador*, que é o último componente da Figura 3.1 a fazer parte do *transmissor*, tem como objetivo transformar os símbolos codificados produzidos pelo codificador de canal em formas de onda apropriadas, para que possam ser transmitidas através do *canal de comunicação*. Este último é qualquer meio físico pelo qual a informação possa ser veiculada, seja na forma de um sinal elétrico, quando o canal será um par de fios condutores; na forma de ondas eletromagnéticas, quando o canal poderá ser o espaço livre ou um guia de ondas; ou na forma de luz, quando o canal poderá ser o espaço livre ou uma fibra óptica; etc. Devemos observar que no canal o sinal transmitido será contaminado com *ruído*. Consideraremos em nosso estudo o ruído aditivo, ou seja, aquele que é adicionado ao sinal. Em comunicações móveis, também há o ruído multiplicativo, que é chamado de *desvanecimento*, mas este não será aqui abordado. Estudaremos com algum detalhe um modelo probabilístico para o ruído aditivo no Capítulo 6. No lado do *receptor*, encontraremos primeiramente o *demodulador*, que receberá como entrada as formas de onda transmitidas porém adicionadas de ruído. O demodulador faz papel inverso do modulador, ou seja, o

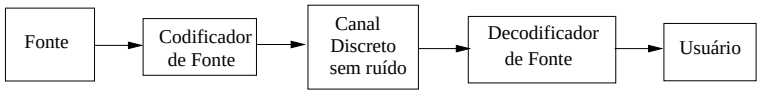


Figura 3.2: Diagrama de blocos para a codificação de fonte.

demodulador transforma formas de onda em símbolos. Obviamente, a partir do exposto acima, o modulador e o demodulador devem ser projetados em conjunto. É importante observar que, devido ao ruído, o demodulador poderá cometer erros e produzir uma seqüência de símbolos diferente daquela que foi transmitida. O papel do *decodificador de canal*, o próximo componente a aparecer na Figura 3.1, é o de detectar e/ou corrigir a maioria desses erros. É também óbvio que o par codificador/decodificador de canal deva ser projetado conjuntamente. Finalmente, o *decodificador de fonte*, que é projetado em conjunto com o codificador de fonte, deverá reintroduzir a informação redundante, recompor a seqüência de símbolos emitida pela fonte e entregá-la ao *usuário final*.

3.2.1 Codificação de Fonte

A fim de tratarmos da codificação de fonte, devemos supor que as seqüências codificadas pela fonte chegarão ao decodificador de fonte sem erro. Ou seja, vamos considerar o diagrama de blocos da Figura 3.2.

De maneira resumida, dizemos que o papel deste codificador é o de retirar a informação redundante (ou inútil, desnecessária) da fonte. Para entender melhor este conceito, suponha que a fonte emita frases da língua portuguesa. O alfabeto da fonte é portanto formado pelas letras A, B, C, etc., bem como alguns caracteres especiais, como por exemplo o espaço em branco. Sem muito esforço, podemos afirmar que esta fonte produz informação redundante. Observe, por exemplo, que na língua portuguesa uma letra "q" é sempre seguida de uma letra "u". Cientes de que a fonte emite frases da língua portuguesa, poderíamos portanto omitir a transmissão ou a armazenagem da próxima letra, que certamente será "u", sempre que a fonte emitir uma letra "q", o que nos pouparia tempo ou espaço e, conseqüentemente, nos proporcionaria uma economia. Em termos estatísticos, resumiremos a situação acima da seguinte maneira. A probabilidade de o próximo símbolo da fonte ser qualquer letra diferente de "u" dado que o

símbolo anterior foi um "q" é zero. O codificador de fonte portanto tentará explorar o conhecimento da *estatística da fonte*, com o propósito de retirar a sua redundância proporcionando assim uma economia.

Se nos aprofundássemos um pouco mais na discussão anterior, poderíamos também falar na redundância baseada no *contexto*, se soubéssemos de que assunto tratassem as frases emitidas pela fonte. Em Teoria de Informação, restringimo-nos a adotar um modelo probabilístico para a fonte no qual precisamos definir a probabilidade de a fonte emitir cada símbolo dado que conhecemos os últimos (digamos) M símbolos emitidos pela fonte. Diremos que esta é uma *fonte de Markov* de ordem M, pois esta situação corresponde justamente à descrição de uma cadeia de Markov, da teoria de processos estocásticos discretos.

O assunto de codificação de fonte não será abordado aqui, apenas o de codificação de canal, o que será feito no Capítulo 4. Entretanto, antes de encerrarmos este assunto, devemos citar o fato histórico de que a primeira construção de um código de fonte foi realizada muito antes de Shannon publicar os seus resultados. Em 1837, o artista e inventor americano Samuel Morse inventou o telégrafo e, associado a este invento, o código Morse. No código Morse, cada letra do alfabeto é representada por uma seqüência formada pelos símbolos "$\cdot$" e "—", levando-se em consideração a estatística da língua inglesa. Obviamente, a transmissão/armazenagem de um "$\cdot$" consome um tempo/espaço muito menor do que a transmissão/armazenagem de um "—". Na língua inglesa, por exemplo, a letra E ocorre com uma freqüência relativa (probabilidade) de aproximadamente 10,3 %, enquanto que para a letra Q esse valor é de 0,08 %. De maneira inteligente, Morse escolheu transmitir as palavras "$\cdot$" e "—,—,$\cdot$,—" para representar as letras E e Q, respectivamente. A idéia de Morse era de que, como a letra E é mais freqüente, a seqüência representando essa letra deveria ser a mais curta possível. Por outro lado, a letra Q, raramente encontrada em um texto na língua inglesa, poderia ter uma representação mais longa. Deste modo, na média e a longo prazo, haverá uma economia com a redução da redundância da fonte. Podemos concluir dizendo que, na codificação de fonte, estamos preocupados em encontrar maneiras eficientes de representar os símbolos da fonte, ou seja, obter um menor número médio de dígitos por símbolo da fonte.

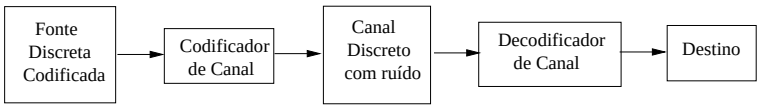


Figura 3.3: Diagrama de blocos para a codificação de canal.

3.2.2 Codificação de Canal

Neste caso há ruído e, conseqüentemente, o que é transmitido nem sempre é recebido sem erros. A fim de tratarmos da codificação de canal, devemos supor que a fonte já tenha sido codificada e que esteja livre de redundância. Ou seja, vamos considerar o diagrama de blocos da Figura 3.3. O papel da codificação de canal é o de minimizar a probabilidade de erro dos símbolos transmitidos através da adição de redundância estruturada à informação a ser transmitida pelo canal ruidoso.

Exemplo 18 *Considere a fonte discreta com alfabeto* $S = \{s_1, s_2, s_3, s_4\}$. *Por simplicidade, denotaremos a fonte também por* S. *Poderíamos associar um par de bits a cada símbolo da fonte, por exemplo,* $s_1 \leftrightarrow 00$, $s_2 \leftrightarrow 01$, $s_3 \leftrightarrow 10$ *e* $s_4 \leftrightarrow 11$. *Adicionando-se um bit de redundância, podemos ter:*

$$\begin{aligned} s_1 &\leftrightarrow 000 \\ s_2 &\leftrightarrow 011 \\ s_3 &\leftrightarrow 101 \\ s_4 &\leftrightarrow 110 \end{aligned}$$

que resulta em um código de canal capaz de detectar 1 erro. Note que se 000 (que corresponde ao símbolo s_1*) for transmitido e houver um único erro de bit no canal, por exemplo, se a seqüência recebida for 010, o decodificador de canal será capaz de reconhecer que houve 1 erro, visto que 010 não é uma seqüência válida. É importante perceber que este código não é capaz de detectar a presença de 2 erros no canal. Por exemplo, se a seqüência recebida for 011, o decodificador de canal dará como resultado que o símbolo transmitido foi o símbolo* s_2*, resultando numa decisão errônea. Este código é, portanto, um código detector de 1 um erro de bit. Vejamos agora um exemplo de um código corretor de erro.*

Exemplo 19 *Considere a fonte $S = \{s_1, s_2\}$ e o código de canal dado por:*

$$s_1 \leftrightarrow 000$$

$$s_2 \leftrightarrow 111$$

Note que se 000 (que corresponde ao símbolo s_1) for transmitido e houver um único error de bit no canal, por exemplo, se a seqüência recebida for 010, o decodificador de canal será capaz de reconhecer que, havendo mais zeros do que uns na seqüência recebida, é mais provável que a seqüência transmitida tenha sido 000, resultando numa decisão correta. Efetivamente, o código terá corrigido 1 erro. Por outro lado, é importante perceber que este código não é capaz de corrigir a presença de 2 erros no canal. Por exemplo, se a seqüência recebida for 011, o decodificador de canal dará como resultado que o símbolo transmitido foi o símbolo s_2, pois haverá mais uns do que zeros na seqüência recebida, levando o decodificador a concluir que a seqüência 111 será seqüencia transmitida mais provável, resultando numa decisão errônea. Portanto, este é um código corretor de 1 um erro de bit.

Em síntese, a teoria da informação, objeto de estudo deste capítulo, nos fornece dois limites fundamentais:

1. **Sobre a codificação de fonte:** O número médio de dígitos por símbolo de uma fonte S é sempre $\geq H(S)$, a *entropia* da fonte.

2. **Sobre a codificação de canal:** Para se ter uma comunicação confiável, ou seja, com uma probabilidade de erro tão pequena quanto se queira, a taxa de transmissão R deve ser $< C$, a *capacidade de canal.*

   Por outro lado, se $R > C$, podemos determinar um limitante inferior para a probabilidade de erro. Ou seja, nenhum sistema de comunicação, usando este canal, poderá operar com uma probabilidade de erro abaixo deste limitante. Portanto, a escolha da probabilidade de erro não poderá ser arbitrária.

3.3 Fontes Discretas sem Memória

Uma *fonte discreta* é qualquer dispositivo que emita seqüências de símbolos pertencentes a um alfabeto fixo e finito, digamos $S = \{s_1, s_2, \ldots, s_n\}$. Para especificar plenamente a fonte discreta, necessitamos de uma distribuição de probabilidade para os símbolos emitidos pela fonte. Vamos supor que os símbolos sucessivos emitidos pela fonte sejam estatisticamente independentes, ou seja,

$$P(s_i^k | s_j^{k-1}) = P(s_i^k)$$

que é lido como: a probabilidade de a fonte emitir o símbolo s_i no instante k dado que o símbolo s_j foi emitido no instante anterior, isto é, no instante $k-1$, é igual à probabilidade de a fonte emitir o símbolo s_i no instante k. Em outras palavras, o símbolo emitido no instante anterior não afeta a probabilidade do símbolo no próximo instante. Neste caso, a fonte é dita ser uma *fonte discreta sem memória*, ou uma DMS, do inglês: *Discrete Memoryless Source*. Em termos de cadeias de Markov, como mencionado na Seção 3.2.1, consideraremos no nosso estudo apenas as fontes de Markov de ordem zero.

Devemos notar que uma DMS é nada mais do que uma variável aleatória discreta.

Exemplo 20 *Poderemos considerar o lançamento de uma moeda como uma fonte discreta sem memória S; faremos as associações "cara" $\leftrightarrow 0$ e "coroa" $\leftrightarrow 1$, e definiremos a fonte como $S = \{0, 1\}$, com $P(0) = P(1) = 1/2$.*

3.4 Medidas de Informação

Vamos considerar uma fonte discreta sem memória como sendo uma variável aleatória discreta X, cujo espaço amostral é também denotado por $X = \{x_1, x_2, \ldots, x_L\}$, com L eventos elementares. Denotaremos por $I(x)$ a quantidade de informação contida no símbolo $x \in X$.

3.4.1 A Medida de Informação de Hartley

Em 1928, Hartley, na tentativa de estabelecer uma medida quantitativa para informação, fez as seguintes considerações:

1. Um símbolo contém informação apenas se houver mais de um valor possível para este símbolo, isto é, se $L \geq 2$. Se $L = 1$, então $I(x) = 0$.

2. Se $L \geq 2$, a quantidade de informação de n símbolos é n vezes a quantidade de informação de um símbolo.

Assim, Hartley propôs:

$$I(x) = \log_b L$$

Note que esta medida de informãção satisfaz as duas condições acima:

1. $L = 1 \Rightarrow I(X) = \log_b 1 = 0$

2. $\underbrace{- - - \ldots -}_{n \text{ símbolos}}$ Para L^n possíveis valores, temos que

$$I(x) = \log_b L^n = n\, I(x)$$

A escolha da base b do logaritmo é arbitrária. Para $b = 2$, a unidade de $I(x)$ é *bits*. Para $b = 10$, a unidade é *Hartley*. E para $b = e = 2,71\ldots$, temos a unidade *nats*.

Exemplo 21 *No caso de uma moeda ($L = 2$), ou seja, X=\{cara, coroa\}, para $b = 2$, temos:*

$$I(x) = \log_2 2 = 1 \textit{ bit de informação}$$

Para $b = e = 2,71\ldots$, temos:

$$I(x) = \log_e 2 = \ln 2 = 0,693 \textit{ nats de informação}$$

O próximo experimento mostra uma falha na medida de Hartley. Considere duas urnas contendo 4 bolas cada, como mostra a Figura 3.4. Seja $X = \{0, 1\}$ uma variável aleatória binária que indica o número inscrito na bola retirada aleatoriamente de uma urna. Note que, como $L = 2$, segundo Hartley, $I(x) = \log_2 2 = 1$ bit para qualquer

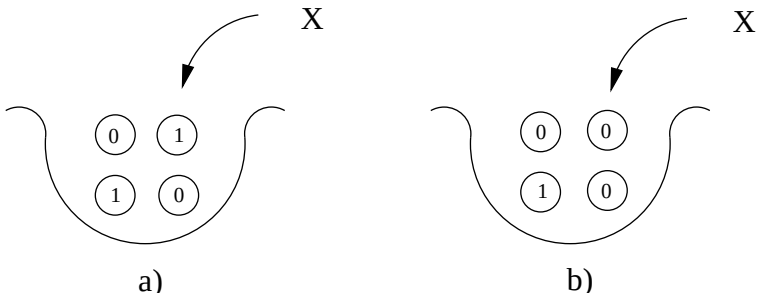


Figura 3.4: Experimento com duas urnas sobre a medida de informação de Hartley.

uma das duas urnas. Porém, intuitivamente, diríamos que a quantidade de informação $I(X = 0)$ obtida quando uma bola 0 é retirada da urna (a) é maior do que quando uma bola 0 é retirada da urna (b). Pensamos desta maneira porque como é muito provável que uma bola retirada da urna (b) seja 0, não nos causa tanta surpresa quando esta é retirada da urna (b). Dizemos então que a quantidade de informação obtida é muito pequena.

3.4.2 A Medida de Informação de Shannon

A medida de Shannon leva em consideração a probabilidade de ocorrência do símbolo. Esta é o ingrediente que faltava na medida de Hartley para satisfazer a nossa intuição, como ilustrado no exemplo das urnas. Shannon definiu a quantidade de informação de um símbolo como:

$$I(x_i) = \log_b \frac{1}{P(X = x_i)} = -\log P(X = x_i)$$

Exemplo 22 *Voltando ao exemplo das urnas, teremos que:*

1. *Na urna (a),* $P(X = 0) = 1/2 \Rightarrow I(0) = \log_2 \frac{1}{\frac{1}{2}} = 1$ *bit*
2. *Na urna (b),* $P(X = 0) = 3/4 \Rightarrow I(0) = \log_2 \frac{1}{\frac{3}{4}} = 0,415$ *bit.*

Portanto, a medida de Shannon satisfaz, além das duas condições de Hartley (verifique isso), a nossa intuição.

Exemplo 23 *Neste exemplo,* $X = \{x_1, x_2, \ldots, x_L\}$ *é uma variável aleatória com distribuição uniforme, ou seja,* $P(x_i) = \frac{1}{L}$*, para* $i = 1, 2, \ldots, L$*. Assim, para todo*

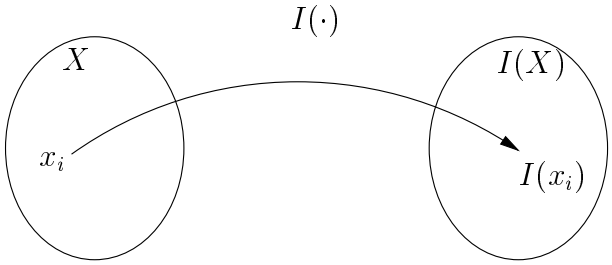


Figura 3.5: Diagrama de Venn para ilustrar a variável aleatória $I(X)$.

$i = 1, 2, \ldots, L$, *temos que* $I(x_i) = \log_b \frac{1}{1/L} = \log_b L$. *Note que neste caso, e somente neste caso, as medidas de Hartley e de Shannon fornecem o mesmo resultado.*

Exemplo 24 *Consideremos agora a variável aleatória* X *tal que* $P(X = x_i) = 1$ *para algum* i, *e* $P(X = x_j) = 0$ *para todo* $j \neq i$. *Assim,* $I(x_i) = \log_b \frac{1}{1} = \log_b 1 = 0$ *e* $I(x_j) = \log_b \frac{1}{0} = \log_b \infty = \infty$.

Devemos perceber a diferença entre um *bit* e um *dígito binário*. Note que o evento 101, com probabilidade 3/4, é composto por 3 dígitos binários, mas contém $0,415$ bit de informação.

3.5 Entropia, Entropia Conjunta e Entropia Condicionada

A seguir, apresentaremos um dos conceitos mais importantes da Teoria da Informação, a saber, a *entropia* de uma variável aleatória. Devemos observar que $I(X)$ é uma função da variável aleatória X, uma vez que para cada evento elementar $x_i \in X$ obtemos um valor $I(x_i)$ a partir da função $I(\cdot)$, como ilustrado na Figura 3.5. Conseqüentemente, $I(X)$ também é uma variável aleatória, com espaço amostral $\{I(x_1), I(x_2), \ldots, I(x_L)\}$, onde $I(x_i)$ ocorre com probabilidade $P(X = x_i)$. No Capítulo 2 vimos como obter a esperança de uma função de uma variável aleatória. Com isso em mente, temos a seguinte definição.

Definição 1 *A entropia (ou incerteza a priori) de uma variável aleatória* X *(ou de*

uma fonte X) é definida como:

$$\begin{aligned} H(X) &\triangleq E\{I(X)\} = \sum_{x_i \in X} P(x_i) I(x_i) \\ &= \sum_{x_i \in X} P(x_i) \log_b \frac{1}{P(x_i)} \end{aligned}$$

$H(X)$ pode ser entendida como a quantidade média de informação produzida por um símbolo da fonte X. Numa outra leitura, $H(X)$ é considerada como a *incerteza a priori* sobre o valor da variável aleatória X.

Antes de se observar X, tem-se uma incerteza a priori igual a $H(X)$ bits. Ao se observar X, tem-se em média uma surpresa de $H(X)$ bits. Depois de X ter sido revelada, ganha-se em média $H(X)$ bits de informação e a incerteza a respeito de X é reduzida a zero.

A partir do exposto no último parágrafo, chegamos a uma importante definição de informação, a saber,

> **"Informação é a diferença entre as incertezas antes e depois da revelação de uma variável aleatória."**

Exemplo 25 *Seja $X = \{x_1, x_2, x_3\}$ uma variável aleatória com $P(x_1) = 1/2$, $P(x_2) = P(x_3) = 1/4$. Assim,*

$$H(X) = \frac{1}{2}\log_2 \frac{1}{1/2} + \frac{1}{4}\log_2 \frac{1}{1/4} + \frac{1}{4}\log_2 \frac{1}{1/4} = 1,5 \text{ } bits$$

Exemplo 26 *Seja N uma variável aleatória de Bernoulli representando o ruído binário num canal binário simétrico, descrita por:*

$$N = \begin{cases} 0, & \text{com probabilidade } 1-p \\ 1, & \text{com probabilidade } p \end{cases}$$

O sistema de comunicação binária correspondente é mostrado na Figura 3.6. A entropia da variável aleatória binária N é obtida por:

$$H(N) = p \log \frac{1}{p} + (1-p) \log \frac{1}{1-p} \triangleq \mathcal{H}(p)$$

A função $\mathcal{H}(p)$ é conhecida como a entropia de uma variável aleatória binária com parâmetro p, e seu gráfico é mostrado na Figura 3.7.

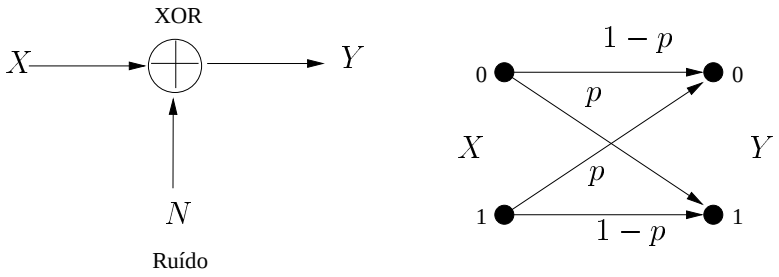


Figura 3.6: Canal binário simétrico com probabilidade de transição p.

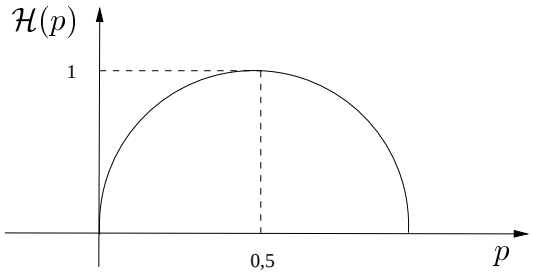


Figura 3.7: Entropia da variável aleatória binária com parâmetro p.

Note que se $N = 0$ ou $N = 1$ com probabilidade 1 (ou seja, se $p = 0$ ou $p = 1$), então não há incerteza, e $\mathcal{H}(0) = \mathcal{H}(1) = 0$. Por outro lado, $p = 0,5$ maximiza a incerteza $\mathcal{H}(p)$, como no caso da moeda equilibrada.

Teorema 1 *Seja X uma variável aleatória com L possíveis valores, segundo uma distribuição de probabilidades $P(x)$. Então,*

$$0 \leq H(X) \leq \log L$$

A igualdade da esquerda ocorre se e somente se $P(x_i) = 1$ para algum i, com $1 \leq i \leq L$, e $P(x_j) = 0$ para todo $j \neq i$. A igualdade da direita ocorre se e somente se $P(x_i) = 1/L$ para todo $1 \leq i \leq L$.

Prova: Para provar que $H(X) \geq 0$, note que como $0 < P(x) \leq 1$ para todo $x \in X$, temos que $P(x)\log(1/P(x)) \geq 0$ para todo $x \in X$. Logo,

$$H(X) = \sum_{x \in X} P(x) \log_b \frac{1}{P(x)} \geq 0.$$

Para provar que $H(X) \leq \log_b L$, precisaremos do lema abaixo. □

Lema 1 (A Desigualdade da Teoria da Informação) *Seja z um número real e*

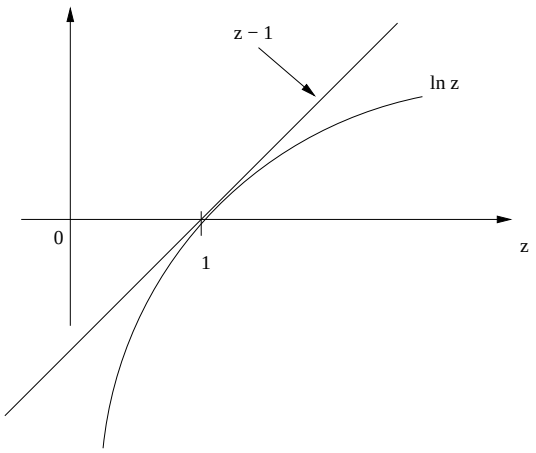


Figura 3.8: Desigualdade da Teoria da Informação.

positivo. Então,

$$\frac{\log_b z}{\log_b e} = \ln z \leq z - 1$$

com igualdade se e somente se $z = 1$.

Prova: A prova pode ser obtida diretamente a partir da Figura 3.8, de onde podemos concluir que

$$\log_b z \leq (z-1)\log_b e.$$

□

Prova: (**Continuação da prova do Teorema 1**) Para provar que $H(X) \leq \log L$, vamos mostrar que $H(X) - \log L \leq 0$, onde por questão de simplicidade não indicaremos a base b do logaritmo. Então,

$$\begin{aligned}
H(X) - \log L &= \sum_x P(x) \log \frac{1}{P(x)} - \log L \left(\sum_x P(x) \right) \\
&= \sum_x P(x) \left[\log \left(\frac{1}{P(x)} \right) - \log L \right] \\
&= \sum_x P(x) \log \left(\frac{1}{L\ P(x)} \right) \\
&\leq \sum_x P(x) \left[\frac{1}{L\ P(x)} - 1 \right] \log e \\
&= \log e - \log e \\
&= 0
\end{aligned}$$

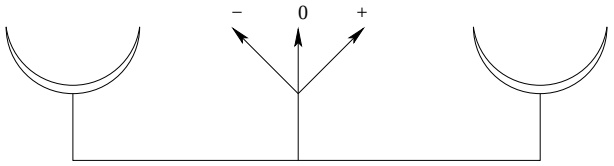


Figura 3.9: Balança para o problema das n bolas com uma bola possivelmente diferenciada.

Note que a desigualdade acima segue do Lema 1, fazendo-se a substituição

$$z = \frac{1}{L\ P(x)}.$$

Assim, está provado que $H(X) \leq \log L$. Do Lema 1, temos a igualdade se e somente se

$$z = \frac{1}{L\ P(x)} = 1 \Rightarrow P(x) = \frac{1}{L}, \forall x \in X$$

ou seja, se e somente se X tiver distribuição uniforme. Isso conclui a prova do Teorema 1. □

Exemplo 27 *Considere um conjunto de n bolas que, aos nossos olhos, têm a mesma aparência e mesmo peso, porém uma delas é possivelmente diferenciada. Ou seja, as n bolas podem ser idênticas ou exatamente uma delas pode ser mais leve ou mais pesada. As três situações para a bola diferenciada são eqüiprováveis. Devemos descobrir a bola diferenciada (se existir) através do uso de uma balança de feira, como indicada na Figura 3.9. Dado o número médio K de pesagens, e supondo que a balança nos forneça 3 possíveis resultados (-, +, ou 0), devemos encontrar um limitante superior para o número de bolas, n, de modo que a bola diferenciada possa ser determinada com este número médio K de pesagens.*

Resolvemos este problema da seguinte maneira. Primeiramente, teremos que calcular a incerteza do problema. Se as n bolas são numeradas de 1 a n, poderemos ter a bola 1 mais leve, ou a bola 1 mais pesada, ou a bola 2 mais leve, ou a bola 2 mais pesada, e assim por diante, até ter a bola n mais leve ou mais pesada. Contando com a possibilidade de as n bolas serem idênticas, teremos $2n + 1$ possíveis situações para as n bolas, e estas são eqüiprováveis. Assim, a incerteza do problema é:

$$H(X) = \log_2(2n + 1) \text{ bits.}$$

Agora devemos obter a quantidade máxima de informação que a balança poderá produzir com uma única pesagem. Note que há apenas 3 possíveis resultados de pesagem. Supondo-se que a balança seja carregada de modo que os 3 resultados sejam eqüiprováveis, e supondo-se que os resultados de sucessivas pesagens sejam estatisticamente independentes, então a quantidade média de informação obtida da balança é:

$$K \log_2 3 \quad \text{bits.}$$

Daí, obtemos

$$K \log_2 3 \geq H(X) = \log_2(2n+1) \Rightarrow n \leq \frac{2^{K \log_2 3} - 1}{2}$$

Para $K = 3$, teremos que $n \leq 13$. De fato, é possível resolver o problema com $n = 12$ bolas e 3 pesagens (tente uma solução). É impossível resolver o problema com 3 pesagens se o número de bolas for 14.

Vamos agora estender o conceito de entropia para mais de uma variável aleatória.

Definição 2 *A* entropia conjunta $H(X,Y)$ *(ou* $H(XY)$*) de um par de variáveis aleatórias* (X,Y) *com distribuição de probabilidades conjunta* $P(x,y)$ *é definida como:*

$$H(X,Y) \stackrel{\Delta}{=} \sum_x \sum_y P(x,y) \log \left[\frac{1}{P(x,y)} \right]$$

Note que não há nada de novo nessa definição, uma vez que (X,Y) pode ser considerado como uma única variável aleatória (na verdade um vetor aleatório) $Z = (X,Y)$. Naturalmente, podemos ter a entropia conjunta para n variáveis aleatórias: $H(X_1, X_2, \ldots, X_n)$.

Como visto na Seção 2.11, podemos também obter a esperança de uma função de uma variável aleatória X condicionada ao evento $Y = y$. A entropia de X condicionada ao evento $Y = y$ é definida a seguir.

Definição 3 *A entropia condicionada de uma variável aleatória* X*, dado que* $Y = y$*, é definida como:*

$$H(X|Y=y) = \sum_{x \in X} P(x|y) \log \frac{1}{P(x|y)}$$

e a entropia condicionada de X, dado Y, como:

$$\begin{aligned} H(X|Y) &= \sum_{y\in Y} P(y)H(X|Y=y) \\ &= \sum_{y\in Y} P(y)\left(\sum_{x\in X} P(x|y)\log\frac{1}{P(x|y)}\right) \\ &= \sum_{x\in X}\sum_{y\in Y} P(x|y)P(y)\log\frac{1}{P(x|y)} \\ &= \sum_{x\in X}\sum_{y\in Y} P(x,y)\log\frac{1}{P(x|y)} \end{aligned}$$

Exemplo 28 *Considere o canal de comunicação binário da Figura 3.6. Digamos que os eventos $X=0$ e $X=1$ sejam eqüiprováveis. Devemos calcular $H(X|Y=y)$ para $y=0$ e $y=1$ e, em seguida, obter $H(X|Y)$.*

Como X é uma variável aleatória binária uniformemente distribuída, teremos que $H(X)=1$ bit. Calculando $H(X|Y=0)$, temos que:

$$H(X|Y=0)=\sum_{x\in X} P(x|0)\log\frac{1}{P(x|0)}$$

Note que não dispomos da probabilidade $P(x|y)$. Mas, pelo teorema de Bayes:

$$\begin{aligned} P(x|y) &= P(X=x|Y=y) \\ &= \frac{P(Y=y|X=x)P(X=x)}{P(Y=y|X=0)P(X=0)+P(Y=y|X=1)P(X=1)} \end{aligned}$$

onde todos as probabilidades envolvidas são conhecidas. Analogamente, podemos calcular $H(X|Y=1)$. Da Definição 3, podemos facilmente calcular $H(X|Y)$.

Teorema 2 *Para quaisquer variáveis aleatórias discretas X e Y,*

$$H(X|Y)\leq H(X)$$

com igualdade se e somente se X e Y forem estatisticamente independentes.

Prova: Devemos provar que $H(X|Y) - H(X) \leq 0$. Assim,

$$\begin{aligned}
H(X|Y) - H(X) &= \sum_x \sum_y P(x,y) \log \frac{1}{P(x|y)} - \sum_x P(x) \log \frac{1}{P(x)} \\
&= \sum_x \sum_y P(x,y) \left[\log \frac{1}{P(x|y)} - \log \frac{1}{P(x)} \right] \\
&= \sum_x \sum_y P(x,y) \log \left(\frac{P(x)}{P(x|y)} \right) \\
&= \sum_x \sum_y P(x,y) \log \left(\frac{P(x)P(y)}{P(x,y)} \right) \\
&\leq \sum_x \sum_y P(x,y) \left[\frac{P(x)P(y)}{P(x,y)} - 1 \right] \log e \\
&= (1-1) \log e \\
&= 0
\end{aligned}$$

onde a desigualdade acima segue do Lema 1, fazendo-se a substituição:

$$z = \frac{P(x)P(y)}{P(x,y)}.$$

Portanto, $H(X|Y) \leq H(X)$. Do Lema 1, a igualdade ocorre se e somente se:

$$\frac{P(x)P(y)}{P(x,y)} = 1$$

Ou seja, se e somente se $P(x,y) = P(x)P(y)$, que equivale a dizer que X e Y são estatisticamente independentes. □

Esse resultado é bastante intuitivo. Note que se X e Y forem estatisticamente independentes, então o conhecimento de Y não deverá reduzir a incerteza a respeito de X. Logo, $H(X|Y) = H(X)$, e ficamos com a mesma incerteza a respeito de X observando Y ou não.

Exemplo 29 *Sejam X e Y variáveis aleatórias com distribuição conjunta $P(x,y)$ dada por $P(1,1) = 0$, $P(1,2) = 1/8$, $P(2,1) = 3/4$ e $P(2,2) = 1/8$. Devemos calcular $H(X)$, $H(X|Y=1)$, $H(X|Y=2)$ e $H(X|Y)$, e comentar os resultados.*

Para calcularmos $H(X)$, precisamos da distribuição de probabilidade marginal $P(x)$:

$$P(x) = \sum_y P(x,y)$$

Assim,

$$P(X=1) = \sum_y P(1,y) = P(1,1) + P(1,2) = 0 + \frac{1}{8} = \frac{1}{8}$$

e

$$P(X=2) = \sum_y P(2,y) = \frac{7}{8}.$$

A entropia $H(X)$ pode agora ser obtida:

$$H(X) = \sum_x P(x) \log_2 \frac{1}{P(x)} = \frac{1}{8} \log_2 8 + \frac{7}{8} \log_2 \left(\frac{8}{7}\right) = 0,544 \text{ bits}$$

A seguir, as entropias condicionadas podem ser obtidas. Precisamos da probabilidade condicionada $P(x|y)$, que pode ser facilmente obtida usando-se a regra de Bayes. Assim,

$$H(X|Y=1) = \sum_x P(x|Y=1) \log_2 \frac{1}{P(x|Y=1)} = 0$$

e

$$H(X|Y=2) = \sum_x P(x|Y=2) \log_2 \frac{1}{P(x|Y=2)} = 1 \text{ bit}$$

Note que, curiosamente, $H(X|Y=2) > H(X)$. Ou seja, uma informação privilegiada aumentou a incerteza sobre X. O conhecimento particular de um evento $Y=y$ pode eventualmente aumentar a incerteza sobre X, mas em média o conhecimento de Y reduz a incerteza sobre X, como demonstramos no Teorema 2, ou seja, *em média, informação privilegiada reduz incerteza.* Note que ao tentar desvendar um crime, um detetive pode ter a sua incerteza aumentada a partir de uma evidência particular. Mas, na média,

"Evidências reduzem incerteza"

Podemos verificar mais uma vez essa afirmação no cálculo de $H(X|Y)$:

$$\begin{aligned} H(X|Y) &= \sum_y P(y) H(X|Y=y) \\ &= \frac{3}{4} \times 0 + \frac{1}{4} \times 1 = 0,25 < H(X) \end{aligned}$$

Teorema 3 (Regra da cadeia para $H(X,Y)$) *Sejam X e Y variáveis aleatórias discretas quaisquer. Então:*

$$H(X,Y) = H(X) + H(Y|X)$$

ou

$$H(X,Y) = H(Y) + H(X|Y)$$

Prova: Escrevemos

$$\begin{aligned}
H(X,Y) &= -\sum_x \sum_y P(x,y) \log P(x,y) \\
&= -\sum_x \sum_y P(x,y) \log \left(P(y|x)P(x)\right) \\
&= -\sum_x \sum_y P(x,y) \log P(x) - \sum_x \sum_y P(x,y) \log(P(y|x)) \\
&= -\sum_x P(x) \log P(x) - \sum_x \sum_y P(x,y) \log(P(y|x)) \\
&= H(X) + H(Y|X)
\end{aligned}$$

Analogamente, podemos mostrar que $H(X,Y) = H(Y) + H(X|Y)$. □

Exemplo 30 *Sejam X e Y variáveis aleatórias discretas descritas pela distribuição de probabilidades conjunta mostrada na tabela abaixo.*

P(x,y)	*X=1*	*X=2*	*X=3*	*X=4*
Y=1	*1/8*	*1/16*	*1/32*	*1/32*
Y=2	*1/16*	*1/8*	*1/32*	*1/32*
Y=3	*1/16*	*1/16*	*1/16*	*1/16*
Y=4	*1/4*	*0*	*0*	*0*

Verificamos que $P(x) = (1/2, 1/4, 1/8, 1/8)$, e que $P(y) = (1/4, 1/4, 1/4, 1/4)$. Assim, $H(X) = 7/4$ bits e $H(Y) = 2$ bits. A entropia conjunta é obtida por:

$$\begin{aligned}
H(X|Y) &= \sum_{i=1}^{4} P(Y=i) H(X|Y=i) \\
&= \frac{1}{4} \sum_{i=1}^{4} H(X|Y=i) \\
&= \frac{1}{4} \Bigg\{ \underbrace{H_{X|Y}(1/2, 1/4, 1/8, 1/8)}_{\text{para } Y=1} + \underbrace{H_{X|Y}(1/4, 1/2, 1/8, 1/8)}_{\text{para } Y=2} +
\end{aligned}$$

$$+ \underbrace{H_{X|Y}(1/4,1/4,1/4,1/4)}_{\text{para } Y=3} + \underbrace{H_{X|Y}(1,0,0,0)}_{\text{para } Y=4} \Bigg\}$$

Calculando as entropias condicionadas, obtemos:

$$\begin{aligned} H(X|Y) &= \frac{1}{4}\left\{\frac{7}{4}+\frac{7}{4}+2+0\right\} \\ &= \frac{11}{8} \text{ bits} \end{aligned}$$

Note que fizemos uso da seguinte relação:

$$P(X=1|Y=1) = \frac{P(X=1,Y=1)}{P(Y=1)} = \frac{1/8}{1/4} = \frac{1}{2}$$

Também, $H(Y|X) = 13/8$ e $H(X,Y) = 27/8$. Assim, constatamos que:

1. $H(X,Y) = 27/8 = H(X) + H(Y|X) = 7/4 + 13/8 = 27/8$
2. $H(X,Y) = 27/8 = H(Y) + H(X|Y) = 2 + 11/8 = 27/8$
3. $H(Y|X) < H(Y) \Leftrightarrow 13/8 < 2 = 16/8$
4. $H(X|Y) \neq H(Y|X) \Leftrightarrow 11/8 \neq 13/8$

Mais adiante, mostraremos que para quaisquer variáveis aleatórias X e Y, temos que $H(X) - H(X|Y) = H(Y) - H(Y|X)$. Podemos verificar esta igualdade com o problema do Exemplo 30:

$$H(X) - H(X|Y) = 7/4 - 11/8 = 3/8 = H(Y) - H(Y|X) = 2 - 13/8 = 3/8$$

Esta medida é denotada por $I(X,Y)$ e é conhecida por *informação mútua*. A informação mútua será formalmente definida mais adiante.

Teorema 4 (Regra da cadeia para n variáveis aleatórias)

$$H(X_1, X_2, \ldots, X_n) = H(X_1) + H(X_2|X_1) + H(X_3|X_1, X_2) + \ldots + H(X_n|X_1, X_2, \ldots, X_{n-1})$$

Prova: Segue facilmente por indução matemática. □

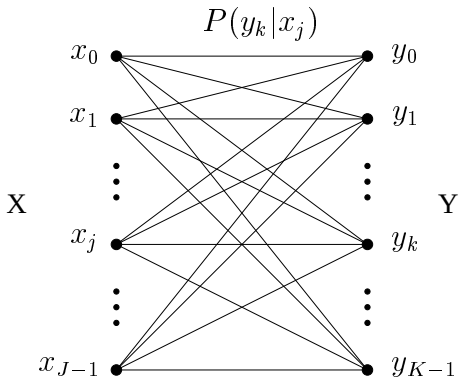


Figura 3.10: Canal discreto sem memória com J entradas e K saídas.

3.6 Canais Discretos sem Memória (DMC)

Os símbolos de uma DMS (fonte discreta sem memória) podem ser transmitidos através de um canal discreto sem memória (DMC, do inglês *Discrete Memoryless Channel*). Um DMC é um canal ruidoso através do qual qualquer um dos J símbolos de $X = \{x_0, x_1, \ldots, x_{J-1}\}$ pode ser transmitido. O símbolo recebido pode ser qualquer um dos K símbolos de $Y = \{y_0, y_1, \ldots, y_{K-1}\}$, segundo a probabilidade condicionada: $P(Y = y_k|X = x_j) = P(y_k|x_j)$, para $i = 0, 1, 2, \ldots, J-1$ e $j = 0, 1, 2, \ldots, K-1$. A representação gráfica deste canal discreto sem memória é mostrada na Figura 3.10, onde:

$$\begin{aligned} X &= \text{alfabeto de entrada} \\ Y &= \text{alfabeto de saída} \\ p(y_k|x_j) &= \text{probabilidade condicionada de } y_k \text{ dado } x_j. \end{aligned}$$

Estes canais são ditos sem memória pois o resultado da transmissão de um símbolo num dado instante não depende do resultado das últimas transmissões, ou seja, a variável aleatória representando o ruído no instante discreto k é estatisticamente independente da variável aleatória representando o ruído no instante discreto $j \neq k$. Daí se dizer que o canal não tem memória.

Exemplo 31 *Considere o canal sem ruído (Figura 3.11): $J = K$*

$$p(y_k|x_j) = \begin{cases} 1, & x_j = y_k \\ 0, & x_j \neq y_k \end{cases}$$

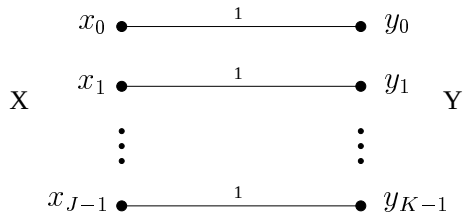


Figura 3.11: Canal sem ruído com J entradas e K saídas.

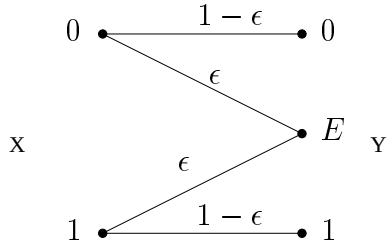


Figura 3.12: Canal binário com apagamento (BEC) com parâmetro ϵ.

Exemplo 32 *O canal binário simétrico (BSC), já apresentado anteriormente na Figura 3.6, também é um exemplo de canal discreto sem memória, onde* $J = K = 2$*, e:*

$$p(y_k|x_j) = \begin{cases} 1-p, & j = k \\ p, & j \neq k \end{cases}$$

Exemplo 33 *Para* $J = 2$ *e* $K = 3$*, temos o canal binário com apagamento (BEC, do inglês* Binary Erasure Channel*), mostrado na Figura 3.12.*

Um DMC é plenamente especificado pelos alfabetos X e Y e pela matriz de canal:

$$P = \begin{bmatrix} P(y_0|x_0) & P(y_1|x_0)\ldots & P(y_{K-1}|x_0) \\ P(y_0|x_1) & P(y_1|x_1)\ldots & P(y_{K-1}|x_1) \\ \vdots & \vdots & \vdots \\ P(y_0|x_{J-1}) & P(y_1|x_{J-1})\ldots & P(y_{K-1}|x_{J-1}) \end{bmatrix}$$

Note que:

$$\sum_{k=0}^{K-1} P(y_k|x_j) = 1, \quad \forall j \in \{1, 2, \ldots, J-1\}$$

Também, dado $P(x_j)$ $\forall j$, temos que

$$P(y_k) = \sum_{j=0}^{J-1} P(y_k|x_j)P(x_j), \quad k = 0, 1, \ldots, K-1$$

Podemos calcular a probabilidade de erro dado que x_j tenha sido transmitido (fazemos $J = K$, e dizemos que houve um erro se y_k for recebido e x_j transmitido para qualquer $k \neq j$):

$$P_{e|x_j} = Prob\{erro|x_j\} = \sum_{\substack{k=0 \\ k\neq j}}^{K-1} P(y_k|x_j) = 1 - P(y_j|x_j)$$

onde $P(y_j|x_j)$ é a probabilidade de acerto.

A probabilidade de erro média é dada por:

$$\begin{aligned} P_e &= \sum_{j=0}^{J-1} P_{e|x_j} P(x_j) \\ &= \sum_{j=0}^{J-1} \sum_{\substack{k=0 \\ k\neq j}}^{K-1} P(y_k|x_j) P(x_j) \end{aligned}$$

A probabilidade de acerto é dada por:

$$P_c = 1 - P_e$$

Exemplo 34 *Para o canal BSC ($J = K = 2$), temos que*

$$\begin{aligned} P_e &= \sum_{k=0}^{1} \sum_{\substack{j=0 \\ j\neq k}}^{1} P(y_k|x_j) P(x_j) \\ &= P(y_0|x_1)P(x_1) + P(y_1|x_0)P(x_0) \\ &= p\, P(x_0) + p\, P(x_1) \\ &= p\,[P(x_0) + P(x_1)] \\ &= p \quad \textit{(canal BSC)} \end{aligned}$$

sectionInformação Mútua

Consideremos o canal discreto sem memória mostrado na Figura 3.13. Estamos interessados nas quantidades:

$$H(X) = \text{ incerteza sobre } X \text{ antes de se observar } Y$$

$$H(X|Y) = \text{ incerteza sobre } X \text{ depois de se observar } Y$$

e na diferença:

$$H(X) - H(X|Y)$$

que é a parte da incerteza sobre X que é "resolvida" ao se observar Y.

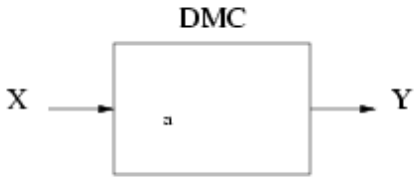


Figura 3.13: Canal discreto sem memória.

Definição 4 *A informação mútua entre duas variáveis aleatórias X e Y, denotada por $I(X,Y)$, é definida como:*

$$I(X,Y) \stackrel{\Delta}{=} H(X) - H(X|Y)$$

É muito importante observar que $I(X,Y)$ representa a quantidade de informação sobre X que obtemos ao observarmos Y. Essa observação passará a ter um significado extramente importante no contexto de comunicações, quando X representar os dados a serem transmitidos por um canal ruidoso e Y representar o sinal recebido na saída deste canal. Neste contexto, $I(X,Y)$ representa a quantidade de informação transmitida pelo canal. Tocaremos neste ponto mais uma vez mais adiante.

Teorema 5 (Simetria)

$$I(X,Y) = I(Y,X)$$

Prova: Para provar este teorema basta desenvolver a equação para $I(X,Y)$ e, após algumas manipulações algébricas, chega-se ao resultado. □

Teorema 6 (Não-negatividade) $I(X,Y) \geq 0$, *com igualdade se e somente se X e Y são estatisticamente independentes.*

Prova: Da Definição 4, e usando o Teorema 2, escrevemos:

$$I(X,Y) = H(X) - H(X|Y) \geq H(X) - H(X) = 0$$

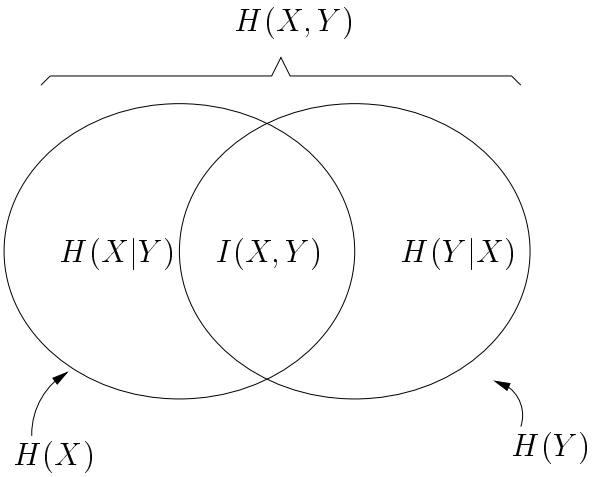


Figura 3.14: Relação entre as entropias e a informação mútua.

com igualdade se e somente se $H(X|Y) = H(Y)$. Mas isso acontece se e somente se X e Y são independentes. □

Teorema 7 $H(X,Y) = H(X) + H(Y) - I(X,Y)$

Prova: Pela regra da cadeia,

$$H(X,Y) = H(X) + H(Y|X)$$

Do Teorema 5, temos que

$$I(X,Y) = I(Y,X) = H(Y) - H(Y|X)$$

ou seja,

$$H(Y|X) = H(Y) - I(X,Y)$$

Assim,

$$\begin{aligned} H(X,Y) &= H(X) + H(Y|X) \\ &= H(X) + H(Y) - I(X,Y) \end{aligned}$$

e o teorema está provado. □

A partir dos resultados acima, é possível resumir as relações entre as entropias e a informação mútua num diagrama de Venn, como indicado na Figura 3.14.

Exemplo 35 *Consideremos mais uma vez o canal BSC, com* $P(X=0) = P(X=1) = 1/2$, *e parâmetro p, como indicado na Figura 3.6. Devemos calcular a informação mútua* $I(X,Y)$. *Podemos usar qualquer uma das duas equações abaixo:*

$$I(X,Y) = H(X) - H(X|Y)$$

ou

$$I(X,Y) = H(Y) - H(Y|X)$$

A segunda equação é mais apropriada. Assim, obtemos $H(Y|X)$*:*

$$H(Y|X) = H(Y|X=0)P(X=0) + H(Y|X=1)P(X=1)$$

Dado que $X=0$, *temos que*

$$P(Y=0|X=0) = 1-p$$

e

$$P(Y=1|X=0) = p$$

Logo,

$$H(Y|X=0) = (1-p)\log\frac{1}{1-p} + p\log\frac{1}{p} \triangleq \mathcal{H}(p)$$

Similarmente, podemos obter

$$H(Y|X=1) = \mathcal{H}(p)$$

Deste modo, temos que

$$H(Y|X) = \mathcal{H}(p)$$

Pelo teorema da probabilidade total, temos que

$$\begin{aligned} P(Y=0) &= P(Y=0|X=0)P(X=0) + P(Y=0|X=1)P(X=1) \\ &= (1-p)\,\frac{1}{2} + p\,\frac{1}{2} \\ &= \frac{1}{2} \end{aligned}$$

Assim,

$$P(Y=0) = P(Y=1) = \frac{1}{2}$$

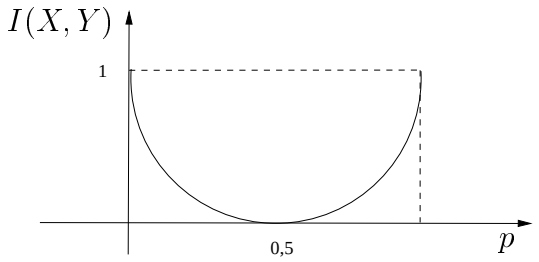


Figura 3.15: Informação mútua para o canal BSC com $P(X=0) = P(X=1) = 1/2$, em função do parâmetro p.

e conseqüentemente:

$$H(Y) = 1 \ \ bit$$

Finalmente, a informação mútua é dada por:

$$I(X,Y) = 1 - \mathcal{H}(p)$$

cujo gráfico em função do parâmetro p do canal BSC é mostrado na Figura 3.15.

Devemos notar que, devido à simetria do canal BSC, temos que $H(Y|X) = \mathcal{H}(p)$, independentemente da distribuição de probabilidade da entrada do canal, ou seja, $P(x)$. Vejamos a seguir o caso do canal BSC com distribuição $P(x)$ não uniforme.

Exemplo 36 *Considere o canal BSC com parâmetro p, e com* $P(X=0) = 1-\gamma$ *e* $P(X=1) = \gamma$. *Neste caso,*

$$I(X,Y) = H(Y) - H(Y|X) = H(Y) - \mathcal{H}(p)$$

Para obtermos $H(Y)$, *devemos calcular:*

$$\begin{aligned} P(Y=0) &= P(Y=0|X=0)P(X=0) + P(Y=0|X=1)P(X=1) \\ &= (1-p)\,(1-\gamma) + p\,\gamma \\ &= 1 - p - \gamma + 2p\gamma \end{aligned}$$

Temos também que

$$P(Y=1) = 1 - P(Y=0) = p + \gamma - 2p\gamma$$

Finalmente, a entropia $H(Y)$ *é dada por:*

$$\begin{aligned} H(Y) &= (1-p-\gamma+2p\gamma)\log\frac{1}{(1-p-\gamma+2p\gamma)} + (p+\gamma-2p\gamma)\log\frac{1}{(p+\gamma-2p\gamma)} \\ &= \mathcal{H}(p+\gamma-2p\gamma) \end{aligned}$$

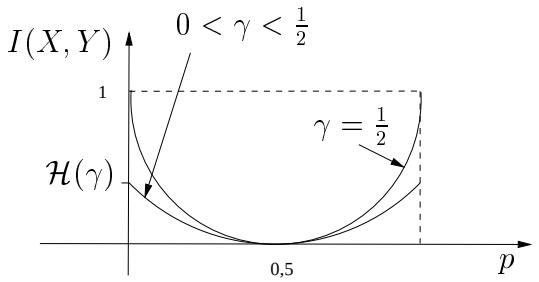


Figura 3.16: Informação mútua em função do parâmetro p para o canal BSC, com $P(X=0) = 1-\gamma$ e $P(X=1) = \gamma$. Note que, para qualquer valor de p, $I(X,Y)$ é máxima para $\gamma = 1/2$.

A informação mútua para o canal BSC com parâmetro p, e com $P(X=0) = 1-\gamma$ *e* $P(X=1) = \gamma$*, é portanto:*

$$I(X,Y) = \mathcal{H}(p+\gamma-2p\gamma) - \mathcal{H}(p)$$

cujos gráficos, para $\gamma = 1/2$ *e para* $0 < \gamma < 1/2$*, são mostrados na Figura 3.16.*

É importante observarmos que $\gamma = 1/2$ maximiza $I(X,Y)$ para qualquer valor fixo de p, ou seja, para qualquer canal BSC fixo. Isto significa que a quantidade máxima de informação que pode ser transmitida pelo canal BSC com parâmetro p é obtida quando a distribuição $P(x)$ for uniforme ($\gamma = 1/2$), ou seja, $P(X=0) = P(X=1) = 1/2$. Isso nos leva ao conceito de capacidade de canal, apresentado na próxima seção.

3.7 Capacidade de Canal de um Canal Discreto Sem Memória - o Caso do Canal (Ruidoso) BSC

Lembremos de que um canal discreto sem memória (ou seja, um DMC) especifica a probabilidade condicionada $P(y|x)$. Porém, a distribuição de probabilidades da entrada do canal, ou seja $P(x)$, pode ser escolhida a fim de maximizar a quantidade de informação transmitida pelo canal, como no caso da seção anterior.

Definição 5 *A capacidade de canal de um canal DMC, denotada por C, é definida por:*

$$C \stackrel{\Delta}{=} \max_{P(x)} I(X,Y) \quad \textit{bits por uso do canal}$$

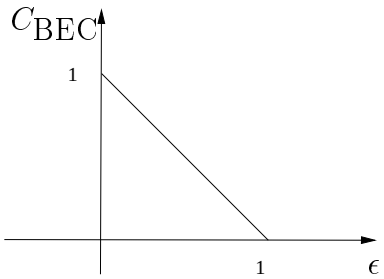


Figura 3.17: Capacidade de canal do canal binário com apagamento (BEC) com parâmetro ϵ.

Equivalentemente,

$$C \triangleq \max_{P(x)} [H(Y) - H(Y|X)]$$

Exemplo 37 *Para o canal BSC com parâmetro p, a partir da Figura 3.16, concluimos que a capacidade de canal é dada por:*

$$C_{BSC} = 1 - \mathcal{H}(p)$$

onde $P(x)$ uniforme (ou seja, $\gamma = 1/2$) maximiza $I(X,Y)$.

Exemplo 38 *Consideremos o canal binário com apagamento (BEC) do Exemplo 32 e ilustrado na da Figura 3.12. Mais adiante iremos demonstrar que a capacidade de canal C_{BEC} do canal BEC é $C_{BEC} = 1 - \epsilon$, mostrada na Figura 3.17.*

É importante ressaltar que nem sempre a distribuição uniforme para X é a distribuição que maximiza $I(X,Y)$. É assim para o canal BSC por causa da sua simetria.

3.8 Teorema da Codificação de Canal (2^o Teorema de Shannon)

Agora que já sabemos como obter a capacidade de canal C de um canal discreto sem memória, vamos apresentar um dos mais importantes resultados obtidos por Shannon. Ele estabelece que C é um limite fundamental para uma comunicação confiável. Antes

disto, apresentaremos alguns conceitos preliminares de codificação de canal, limitando-nos ao caso binário.

Considere o diagrama de blocos para a codificação de canal mostrado na Figura 3.3. Vamos admitir que a fonte binária produza bits u (com probabilidades $P(u=0) = P(u=1) = 1/2$) e que blocos $(u_1, u_2, \ldots, u_k)$ de k bits sejam formados e apresentados ao codificador de canal. Note que, como u é uniformemente distribuída, temos que $H(u) = 1$ bit, e assim 1 bit equivale a um dígito binário. O codificador de canal, por sua vez, produz uma palavra binária $(v_1, v_2, \ldots, v_n)$ com n bits para cada bloco de informação. Essas são chamadas de *palavras código*. Assim, serão 2^k palavras código. O conjunto de todas as palavras código é chamado de *código de bloco*. Assim, um código de bloco binário é qualquer subconjunto com 2^k n-uplas binárias do conjunto formado por todas as 2^n n-uplas binárias possíveis. O canal discreto sem memória a ser considerado é o canal BSC. Note que, como k bits de informação são transmitidos através de n "usos do canal", a taxa de transmissão, R, é dada por:

$$R = \frac{k}{n} \text{ bits/uso}$$

É importante ressaltar que, na apresentação do próximo resultado, nos restringiremos à transmissão binária via um canal BSC, embora o conceito de capacidade de canal a ser apresentado seja válido para qualquer canal.

Teorema 8 (Teorema da Codificação de Canal ou 2^o Teorema de Shannon) *Se os bits de informação de uma fonte binária forem transmitidos a uma taxa R (em bits por uso do canal) através de um canal BSC com capacidade C (em bits por uso do canal), então o seguinte é dito a respeito da probabilidade de erro.*

1. *Seja P_b a probabilidade de erro de bit. Então,*

$$P_b \geq \mathcal{H}^{-1}(1 - C/R), \quad se\ R > C$$

onde $\mathcal{H}^{-1}(\alpha)$ é o valor de p, com $0 \leq p \leq 1/2$, tal que $\mathcal{H}(p) = p\log\frac{1}{p} + (1-p)\log(\frac{1}{1-p})$ seja igual a α.

Ou seja, se $R > C$, então não existe nenhum esquema de codificação de canal que proporcione uma probabilidade de erro menor do que um certo valor mínimo.

2. *Seja P_E a probabilidade de erro de palavra código. Então,*

$$P_E < 2^{-n(C-R)} + O(1/n), \quad se\ R < C$$

onde $O(1/n)$ representa uma expressão que tende a zero quando n tende a infinito. Ou seja, se $R < C$, então P_E pode ser escolhida tão pequena quanto se queira. Basta usarmos um código de bloco com comprimento de palavra código n muito grande.

Exemplo 39 *Suponha que um BSC tenha capacidade $C = 1/4$ bit/uso do canal, e que a taxa de transmissão seja $R = 1/2 > C$ bit/uso do canal. Então:*

$$\begin{aligned} P_b &\geq \mathcal{H}^{-1}\left(1 - \frac{1/4}{1/2}\right) \\ &= \mathcal{H}^{-}1(1/2) \\ &= 0,11 \end{aligned}$$

Assim, não existe uma maneira de se transmitir a uma taxa $R = 1/2$ bit por uso do canal por este canal obtendo-se uma probabilidade de erro de bit inferior a 0,11.

Exemplo 40 *Suponha agora que um BSC tenha capacidade $C = 1/2$ bit/uso do canal, e que a taxa de transmissão seja $R = 1/4 < C$ bit/uso do canal. Considere também a exigência de que a probabilidade de erro de palavra código satisfaça a desigualdade:*

$$P_E < 10^{-10}$$

ou seja, em média haja uma palavra código errada a cada 10 trilhões de palavras código transmitidas. Então, segundo Shannon (e ignorando o termo $O(1/n)$), existe pelo menos um código tal que

$$P_E < 2^{-\frac{n}{4}} < 10^{-10} \Rightarrow n \geq 133$$

Ou seja, existe um código de bloco com comprimento de palavra código $n = 133$ que satisfaz as exigências do problema.

A prova do Teorema 8 pode ser encontrata em livros de Teoria da Informação. Por ser bastante longa e por envolver conhecimentos que vão além do material apresentado nesta apostila, a prova será omitida.

Exemplo 41 *Calcule a capacidade de canal do canal binário com apagamento (BEC) com parâmetro ϵ mostrado na Figura 3.12.*

Devemos inicialmente calcular a informação mútua $I(X,Y) = H(Y) - H(Y|X)$. A distribuição de probabilidades da entrada do canal, representada pela variável aleatória X, é dada por $P(X=0) = \gamma$ e $P(X=1) = 1-\gamma$. Temos que

$$\begin{aligned} P(Y=0) &= P(Y=0|X=0)P(X=0) + P(Y=0|X=1)P(X=1) \\ &= (1-\epsilon)(1-\gamma) = 1-\epsilon-\gamma+\epsilon\gamma, \end{aligned}$$

$$\begin{aligned} P(Y=1) &= P(Y=1|X=0)P(X=0) + P(Y=1|X=1)P(X=1) \\ &= (1-\epsilon)\gamma = \gamma - \epsilon\gamma \end{aligned}$$

e

$$\begin{aligned} P(Y=E) &= 1 - P(Y=0) - P(Y=1) \\ &= 1 - 1 + \epsilon + \gamma - \epsilon\gamma - \gamma + \epsilon\gamma = \epsilon \end{aligned}$$

Assim

$$H(Y) = (1-\epsilon-\gamma+\epsilon\gamma)\log_2\left(\frac{1}{1-\epsilon-\gamma+\epsilon\gamma}\right) + (\gamma-\epsilon\gamma)\log_2\left(\frac{1}{\gamma-\epsilon\gamma}\right) + \epsilon\log_2\left(\frac{1}{\epsilon}\right)$$

Temos também que,

$$H(Y|X=0) = H(Y|X=1) = H(Y|X) = \epsilon\log_2\left(\frac{1}{\epsilon}\right) + (1-\epsilon)\log_2\left(\frac{1}{1-\epsilon}\right) = \mathcal{H}(\epsilon)$$

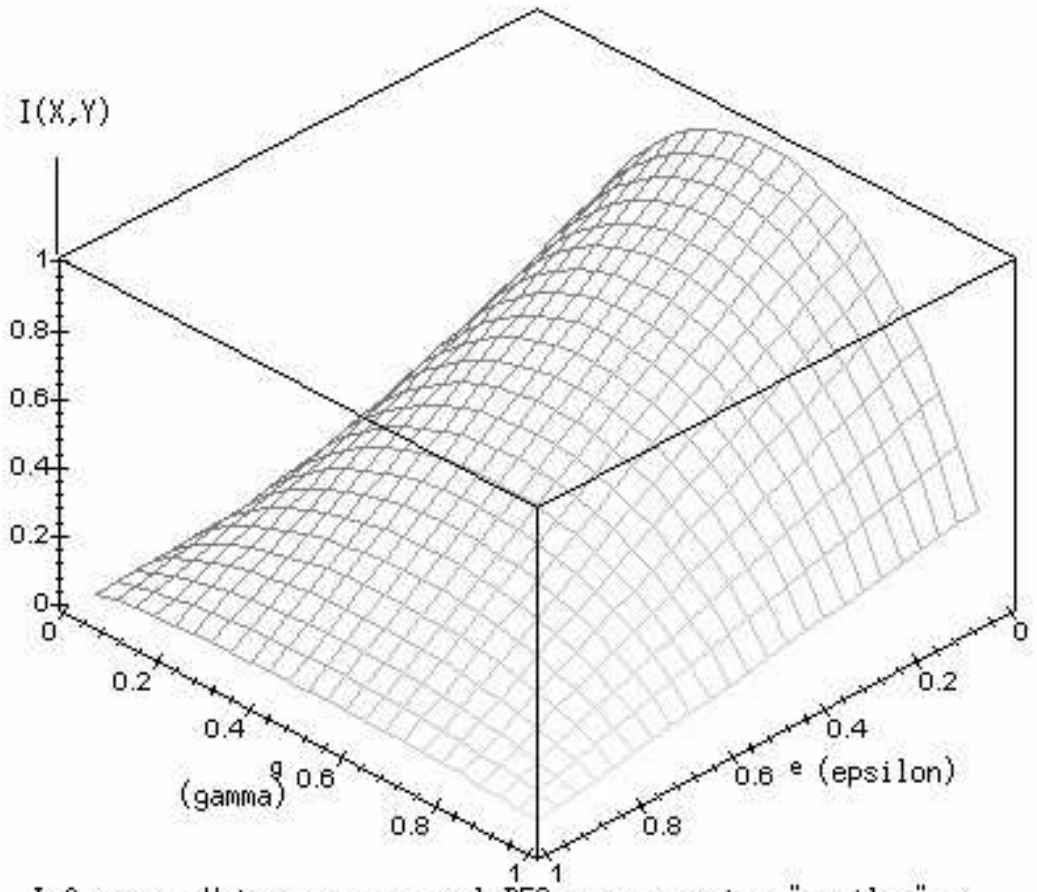
Assim, temos que a informação mútua é dada por:

$$I(X,Y) = H(Y) - \mathcal{H}(\epsilon),$$

onde $H(Y)$ foi obtida acima.

O gráfico de $I(X,Y)$ para o canal BEC em função do parâmetro ϵ do canal e do parâmetro γ, que define a distribuição de probabilidades da entrada do canal, é mostrado na Figura 3.8

Devemos notar que para qualquer valor de ϵ, isto é, para qualquer canal BEC, o valor de γ que maximiza a função $I(X,Y)$ é $\gamma = 1/2$, ou seja, a distribuição de



Informacao Mutua para o canal BEC com parametro "epsilon" e
com distribuicao de entrada P(X=0) = 1 - gamma, e P(X=1) = gamma.

probabilidades da entrada do canal que maximiza $I(X,Y)$ é a distribuição uniforme. Substituindo $\gamma = 1/2$ nas equações acima, ficamos com

$$\begin{aligned} H(Y) &= \frac{1}{2}(1-\epsilon)\log_2\left(\frac{2}{1-\epsilon}\right) + \frac{1}{2}(1-\epsilon)\log_2\left(\frac{2}{1-\epsilon}\right) + \epsilon\log_2\left(\frac{1}{\epsilon}\right) \\ &= (1-\epsilon)\log_2(2) + (1-\epsilon)\log_2\left(\frac{1}{1-\epsilon}\right) + \epsilon\log_2\left(\frac{1}{\epsilon}\right) \\ &= 1-\epsilon+\mathcal{H}(\epsilon) \end{aligned}$$

Finalmente, a capacidade de canal do canal BEC com parâmetro ϵ é dada por:

$$C = \max_{P(x)}\{H(Y) - \mathcal{H}(\epsilon)\} = 1-\epsilon+\mathcal{H}(\epsilon) - \mathcal{H}(\epsilon) = 1-\epsilon$$

O gráfico desta capacidade de canal em função do parâmetro ϵ do canal foi mostrado na Figura 3.17.

Devemos notar que a capacidade obtida, $C = 1-\epsilon$, faz bastante sentido, pois no canal BEC uma fração ϵ da informação é perdida na forma de um apagamento.

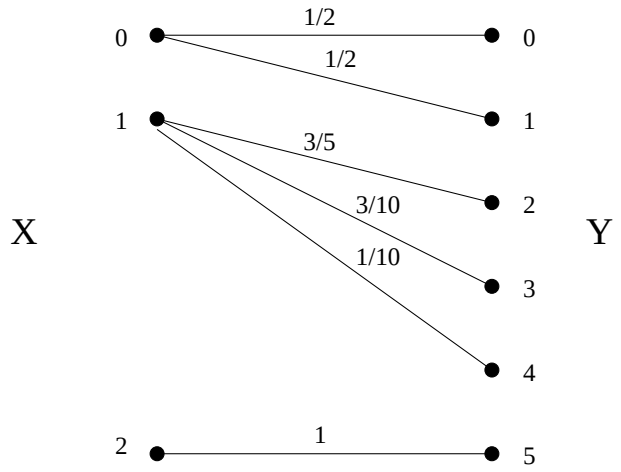


Figura 3.18: Capacidade discreto sem memória.

PROBLEMAS:

1. Considere o canal binário ($J = K = 2$) não simétrico definido por $P(Y = 0|X = 0) = 2/3$, $P(Y = 1|X = 0) = 1/3$, $P(Y = 0|X = 1) = 1/10$ e $P(Y = 1|X = 1) = 9/10$. Faça a representação gráfica deste canal. Supondo que $P(X = 0) = 3/4$ e $P(X = 1) = 1/4$, encontre:

   (a) $P(y)$

   (b) $P(x|y)$

   (c) $H(X)$

   (d) $H(X|Y = 0)$ e $H(X|Y = 1)$

   (e) $H(X|Y)$

   (f) $I(X, Y)$

   Comente os resultados.

2. Calcule a capacidade de canal do canal da Figura 3.18. (**DICA:** Use $I(X, Y) = H(X) - H(X|Y)$).

3. Calcule a capacidade de canal do canal da Figura 3.19. (**DICA:** Use $I(X, Y) = H(Y) - H(Y|X)$).

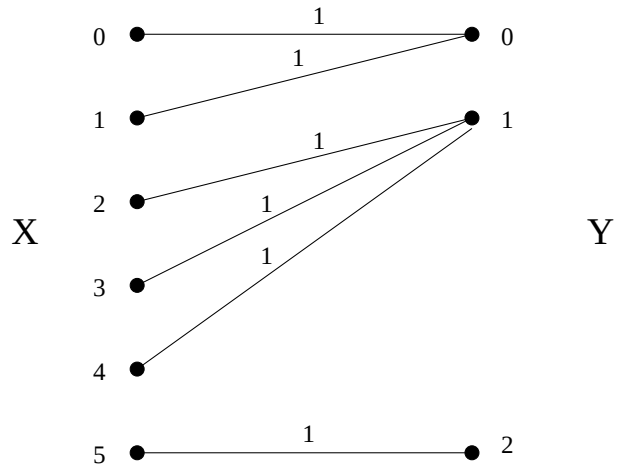


Figura 3.19: Capacidade discreto sem memória.

Capítulo 4

Introdução aos Códigos Corretores de Erro

Como vimos no Capítulo 3, a chave do sucesso para uma comunicação confiável é codificar os bits a serem transmitidos. Neste capítulo, estudaremos alguns conceitos fundamentais da Teoria de Codificação Algébrica e apresentaremos algumas classes de códigos. Os códigos corretores de erro podem ser divididos em duas grandes classes: a dos Códigos de Bloco e a dos Códigos Convolucionais. Na próxima seção, estudaremos os códigos de bloco.

4.1 Códigos de Bloco

O diagrama de blocos de um sistema de comunicação binária com um codificador de bloco, o canal (com ruído) BSC e o decodificador de bloco é mostrado na Figura 4.1. A seqüência de informação produzida pela fonte binária é subdividididade em blocos

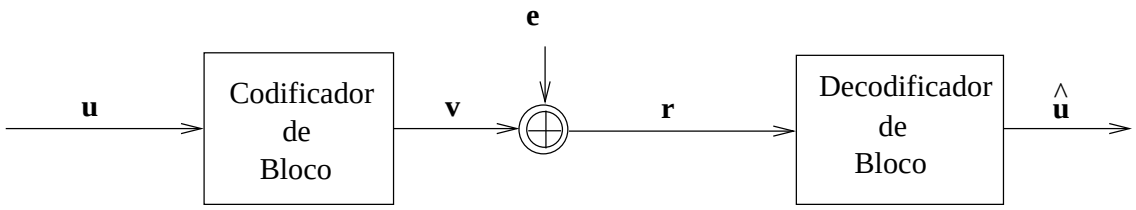


Figura 4.1: Sistema de comunicação binária com código de bloco.

de k bits cada, que serão denotados pelo vetor $\mathbf{u} = (u_0, u_1, \ldots, u_{k-1})$. Como cada $u_i \in \{0, 1\}$, $0 \leq i \leq k-1$, existem 2^k possíveis vetores de informação. A cada um desses vetores, o codificador de bloco associará um vetor binário $\mathbf{v} = (v_0, v_1, \ldots, v_{n-1})$ com n bits, onde $n > k$. O vetor $\mathbf{v}$ será chamado de palavra-código. Como cada vetor de informação é associado a uma única palavra-código, teremos exatamente 2^k palavras-código. O conjunto dessas 2^k palavras-código é chamado de código de bloco. Note que poderíamos ter 2^n possíveis vetores binários de comprimento n. Mas apenas 2^k vetores (justamente as palavras-código) são utilizados pelo codificador. Portanto, um código de bloco é nada mais que um subconjunto C do conjunto de todas os 2^n vetores binários.

O canal que iremos considerar é o canal binário simétrico, ou BSC. Como vimos no Capítulo 3, este canal transmite 1 dígito binário por vez. Como a palavra-código é formada por n dígitos binários, o canal BSC deverá ser usado n vezes para a transmissão de uma palavra-código. Devemos notar se a fonte binária transmite 0's e 1's com igual probabilidade, então cada dígito binário produzido pela fonte contém 1 bit de informação. Como o vetor de informação é formado por k dígitos binários, e a transmissão da palavra-código correspondente usa o canal n vezes, então a taxa de transmissão é $R = k/n$ bits por uso do canal.

Como sabemos, o canal BSC introduz ruído. Há uma probabilidade p de um 0 transmitido ser convertido em um 1, ou de um 1 ser convertido em um 0. Quando há a inversão de um dígito binário no canal, podemos dizer que o ruído 1 foi adicionado (módulo 2) ao dígito binário transmitido. Por exemplo, se transmitirmos o dígito binário $v = 1$ e houver ruído, dizemos que o sinal erro $e = 1$ foi adicionado (módulo 2) ao dígito transmitido, produzindo o dígito recebido $r = v \oplus e = 1 \oplus 1 = 0$. A operação $\oplus$ é também chamada de "ou exclusivo", ou seja, $0 \oplus 0 = 0$, $0 \oplus 1 = 1$, $1 \oplus 0 = 1$ e $1 \oplus 1 = 0$.

Considerando mais uma vez a Figura 4.1, podemos modelar o canal ruidoso pela adição de um vetor erro $\mathbf{e} = (e_0, e_1, \ldots, e_{n-1})$ à palavra-código transmitida $\mathbf{v} = (v_0, v_1, \ldots, v_{n-1})$. Assim, o vetor recebido, que é uma versão ruidosa do vetor

transmitido, é dado por

$$\begin{aligned}\mathbf{r} &= \mathbf{v} \oplus \mathbf{e} \\ &= (v_0, v_1, \ldots, v_{n-1}) \oplus (e_0, e_1, \ldots, e_{n-1}) \\ &= (v_0 \oplus e_0, v_1 \oplus e_1, \ldots, v_{n-1} \oplus e_{n-1}) \\ &= (r_0, r_1, \ldots, r_{n-1}).\end{aligned}$$

O papel do decodificador de canal, também mostrado na Figura 4.1, é obter uma estimativa $\hat{\mathbf{u}}$ para o vetor de informação $\mathbf{u}$, ou equivalentemente, uma estimativa $\hat{\mathbf{v}}$ para a palavra-código transmitida $\mathbf{v}$. Como isto é feito, veremos mais adiante. Vejamos agora um exemplo de um código de bloco.

Exemplo 42 *Vamos considerar um código de bloco de taxa $R = 2/3$, ou seja, com $k = 2$ e $n = 3$. Os possíveis vetores de informação são:* $\mathbf{u}= (0,0), (0,1), (1,0), (1,1)$. *A regra de codificação é a adição de um bit de paridade a cada vetor de informação, de modo que a palavra-código obtida tenha um número par de 1's. Por esta razão, o código abaixo é chamado de código de verificação de paridade, ou simplesmente, código de paridade. Note que podemos ter as seguintes triplas binárias:*

$$000 \;\; 001 \;\; 010 \;\; 011 \;\; 100 \;\; 101 \;\; 110 \;\; 111$$

Escolhemos portanto as seguintes triplas para formar o código de bloco:

$$000 \;\; 011 \;\; 101 \;\; 110$$

Exemplo 43 *Consideremos agora um código de bloco de taxa $R = 1/5$. Os possíveis vetores de informação são:* $\mathbf{u}$*=(0) e (1). Dos $2^5 = 32$ possíveis vetores binários, escolhemos os seguinte vetores para formar o código de bloco: 00000 e 11111. Por razões óbvias, este código é chamado de código de repetição.*

É importante observar que, de um modo geral, para especificarmos um código de bloco de taxa $R = k/n$, precisamos escolher 2^k palavras-código dentre os 2^n possíveis vetores de comprimento n. Isso é uma tarefa muito difícil, pois há $\binom{2^n}{2^k}$ maneiras de se escolher 2^k palavras-código. Em outras palavras, saber qual a melhor escolha, aquela

que proporcione a menor probabilidade de erro, é uma tarefa muito difícil. Precisamos então estabelecer critérios para a escolha de um código. Na próxima seção, veremos o critério de máxima verossimilhança, que minimiza a probabilidade de erro de palavra-código quando os bits da fontes são eqüiprováveis.

4.1.1 Decodificação de Máxima Verossimilhança e a Distância de Hamming

Vamos supor que $\mathbf{u}$ seja o vetor de informação e que a palavra $\mathbf{v}$ tenha sido transmitida. Considerando mais uma vez o diagrama da Figura 4.1, o decodificador deverá estimar $\hat{\mathbf{u}}$, ou $\hat{\mathbf{v}}$, a partir da observação $\mathbf{r}$. Dizemos que um erro de palavra-código acontece quando $\hat{\mathbf{u}} \neq \mathbf{u}$ ou, equivalentemente, quando $\hat{\mathbf{v}} \neq \mathbf{v}$. Denotemos por E o evento erro de palavra-código. Assim, a probabilidade de erro de palavra-código dado que o vetor recebido foi $\mathbf{r}$ é dada por:

$$Prob\{E|\mathbf{r}\} \stackrel{\Delta}{=} Prob\{\hat{\mathbf{v}} \neq \mathbf{v}|\mathbf{r}\}$$

A probabilidade de erro de palavra-código média é dada por:

$$Pr\{E\} = \sum_{\mathbf{r}} Pr\{E|\mathbf{r}\} \times P(\mathbf{r})$$

O decodificador deverá se empenhar para que a estimativa $\hat{\mathbf{v}}$ seja aquela que minimize a probabilidade condicionada de erro $Pr\{E|\mathbf{r}\}$.

Por outro lado, minimizar $Pr\{\hat{\mathbf{v}} \neq \mathbf{v}|\mathbf{r}\}$ é equivalente a maximizar a probabilidade $Pr\{\hat{\mathbf{v}} = \mathbf{v}|\mathbf{r}\}$. Usando a regra de Bayes, podemos reescrever a probabilidade $Pr\{\mathbf{v}|\mathbf{r}\}$ da seguinte maneira:

$$\begin{aligned} P(\mathbf{v}|\mathbf{r}) &= \frac{P(\mathbf{v},\mathbf{r})}{P(\mathbf{r})} \\ &= \frac{P(\mathbf{r}|\mathbf{v})P(\hat{\mathbf{v}})}{P(\mathbf{r})} \end{aligned} \tag{4.1}$$

Supondo-se que os 2^k vetores de informação $\mathbf{u}$ sejam eqüiprováveis, tem-se que as 2^k palavras-código também são eqüiprováveis: $P(\mathbf{v}) = \frac{1}{2^k}$, ou seja, $P(\mathbf{v})$ não depende

de $\mathbf{v}$. Por outro lado, $P(\mathbf{r})$ na equação (4.1) também não depende de $\mathbf{v}$. Assim, maximizar a equação (4.1) é equivalente a maximizar:

$$P(\mathbf{r}|\mathbf{v})$$

Esta é conhecida como a função de verossimilhança, pois é uma medida de quão verossímil (ou parecido com o verdadeiro) é o vetor $\mathbf{r}$. Como estamos considerando um canal sem memória, o dígito recebido r_i não depende estatisticamente do dígito transmitido v_j, para $i \neq j$; r_i depende apenas de v_i. Note que r_i pode ser diferente de v_i, pois o canal é ruidoso. Assim, queremos maximizar a seguinte expressão:

$$\begin{aligned} P(\mathbf{r}|\mathbf{v}) &= P[(r_0, r_1, \ldots, r_{n-1})|(v_0, v_1, \ldots, v_{n-1})] \\ &= \prod_{i=0}^{n-1} P(r_i|v_i) \end{aligned} \tag{4.2}$$

Para o canal BSC temos as seguintes probabilidades condicionadas:

$$P(r_i|v_i) = \begin{cases} 1-p & \text{para } r_i = v_i \\ p & \text{para } r_i \neq v_i \end{cases} \tag{4.3}$$

Para dois vetores $\mathbf{v}$ e $\mathbf{v}'$ quaisquer, definimos a distância de Hamming entre $\mathbf{v}$ e $\mathbf{v}'$ como:

$$d_H(\mathbf{v}, \mathbf{v}') = \text{ número de posições em que } \mathbf{v} \text{ e } \mathbf{v}' \text{ diferem}$$

Exemplo 44 *Considere os vetores binários* $\mathbf{v} = (1, 1, 0, 1, 0)$ *e* $\mathbf{v}' = (1, 0, 0, 0, 1)$. *A distância de Hamming entre* $\mathbf{v}$ *e* $\mathbf{v}'$ *é* $d_H(\mathbf{v}, \mathbf{v}') = 3$.

Agora podemos reescrever a equação (4.2) fazendo uso da equação (4.3) e da definição de distância de Hamming. Assim,

$$\begin{aligned} P(\mathbf{r}|\mathbf{v}) &= \prod_{i=0}^{n-1} P(r_i|v_i) \\ &= p^{d_H(\mathbf{r},\mathbf{v})} \times (1-p)^{n-d_H(\mathbf{r},\mathbf{v})} \\ &= \left(\frac{p}{1-p}\right)^{d_H(\mathbf{v},\mathbf{r})} \times (1-p)^n \end{aligned} \tag{4.4}$$

Note que o termo $(1-p)^n$ não depende de $\mathbf{v}$. Logo, maximizar (4.4) é equivalente a maximizar a seguinte expressão:

$$\left(\frac{p}{1-p}\right)^{d_H(\mathbf{v},\mathbf{r})} \tag{4.5}$$

Como normalmente $p < 1/2$, temos que

$$\left(\frac{p}{1-p}\right) < 1$$

e concluimos que maximizar (4.5) é equivalente a escolher a palavra-código $\mathbf{v}$ que minimize a distância de Hamming entre $\mathbf{v}$ e $\mathbf{r}$, ou seja, $d_H(\mathbf{v}, \mathbf{r})$.

Podemos resumir o processo de decodificação de máxima verossimilhança da seguinte maneira:

Decodificação de Máxima Verossimilhança para Códigos de Bloco

1. Observar a saída do canal $\mathbf{r}$
2. Calcular $d_H(\mathbf{v}, \mathbf{r})$ para todas as 2^k palavras-código do código C
3. Escolher a palavra $\hat{\mathbf{v}}$ mais próxima de $\mathbf{r}$ como a estimativa da palavra-código transmitida, ou seja, escolher $\hat{\mathbf{v}} \in C$ tal que $d_H(\hat{\mathbf{v}}, \mathbf{r}) \leq d_H(\mathbf{v}, \mathbf{r})$ para qualquer $\mathbf{v} \neq \hat{\mathbf{v}} \in C$

Exemplo 45 *Considere o código de taxa $R = 1/5$ do Exemplo 43. Vamos supor que o vetor recebido seja $\mathbf{r} = (1,0,0,1,1)$. Realizando a decodificação de máxima verossimilança, devemos obter primeiramente as distâncias de Hamming*

$$d_H((0,0,0,0,0),(1,0,0,1,1)) = 3 \quad e \quad d_H((1,1,1,1,1),(1,0,0,1,1)) = 2.$$

Note que a palavra-código $\hat{\mathbf{v}} = (1,1,1,1,1)$ está mais próxima de $\mathbf{r}$ do que a palavra-código (0,0,0,0,0). Logo, a palavra decodificada, ou seja, dada como correta, é a palavra $\hat{\mathbf{v}} = (1,1,1,1,1)$.

Devemos notar que para minimizar a probabilidade de erro devemos escolher o código de modo que as distâncias de Hamming entre as palavras-código sejam as maiores possíveis. Desse modo será mais difícil ou mais improvável que uma palavra-código errada (não transmitida) seja confundida pela palavra-código correta (ou transmitida). Podemos visualizar este argumento geométrico com o próximo exemplo.

Exemplo 46 *Considere o código de bloco C de taxa $R = 1/3$ cujas palavras-código são (0,0,0) e (1,1,1). O conjunto de todas as $2^3 = 8$ triplas binárias é mostrado na Figura*

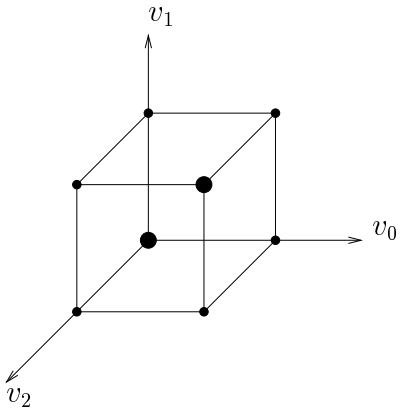


Figura 4.2: Espaço tridimensional contendo as 8 triplas binárias (pontos) possíveis e as 2 palavras-código (pontos grandes) do código de taxa $R = 1/3$ do Exemplo 46.

4.2. As duas palavras-código são representadas pelos pontos (vértices) maiores do cubo tridimensional, enquanto que as triplas não escolhidas para serem palavras-código são representadas pelos pontos (vértices) menores. Note que a distância entre duas triplas quaisquer é maxima se essas triplas são as duas palavras-código.

Como citado anteriormente, a escolha do melhor código dentre todos os possíveis códigos é uma tarefa difícil. Para facilitar esta tarefa devemos impor restrições algébricas ao código. Com essas restrições, alem de reduzirmos o número de possíveis códigos e consequentemente simplificarmos a busca pelo melhor código, também tornaremos viáveis a implementação prática dos codificadores e decodificadores com circuitos lógicos ou seqüenciais. Uma das mais importantes restrições que se pode impor a um código é a linearidade. Na próxima seção, descreveremos a classe dos códigos de bloco lineares.

4.1.2 Códigos de Bloco Lineares

Um código de bloco de taxa $R = k/n$, ou seja, com 2^k palavras-código de comprimento n bits é dito ser um código *linear* se e somente se suas 2^k palavras-código formam um subespaço linear de dimensão k do espaço de todas as n-uplas binárias. Ou seja, um código de bloco binário C é linear se e somente se para quaisquer palavras-código $\mathbf{v} \in C$ e $\mathbf{v}' \in C$ tivermos que $\mathbf{v} \oplus \mathbf{v}' = \mathbf{v}'' \in C$.

Exemplo 47 *Como exemplo de um código linear, consideremos o código de bloco de*

Tabela 4.1: Código de Hamming de Taxa $R = 4/7$.

u	**v**
0000	0000000
1000	1101000
0100	0110100
1100	1011100
0010	1110010
1010	0011010
0110	1000110
1110	0101110
0001	1010001
1001	0111001
0101	1100101
1101	0001101
0011	0100011
1011	1001011
0111	0010111
1111	1111111

taxa $R = 2/3$ *cujas palavras-código são* $\{(0,0,0),(0,1,1),(1,0,1),(1,1,0)\}$. *Podemos verificar facilmente que a soma de quaisquer duas palavras-código resulta numa outra palavra-código. Por exemplo,* $(0,1,1) \oplus (1,1,0) = (1,0,1)$.

Exemplo 48 *Vamos considerar o código de bloco linear de taxa* $R = 4/7$ *mostrado na Tabela 4.1. Este código é conhecido com o código de Hamming de taxa* $R = 4/7$. *Para quaisquer palavras-código, digamos* $\mathbf{v} = (1,1,0,1,0,0,0)$ *e* $\mathbf{v}' = (1,0,1,0,0,0,1)$, *temos que a soma*

$$\mathbf{v} \oplus \mathbf{v}' = (1,1,0,1,0,0,0) \oplus (1,0,1,0,0,0,1) = (0,1,1,1,0,0,1)$$

também é uma palavra-código.

4.1.3 A Matriz Geradora

Uma conseqüência da linearidade do código é que podemos sempre escolher k palavras-código linearmente independentes, formando uma base de um espaço linear k-dimensional, de tal modo que qualquer palavra-código possa ser escrita como uma combinação linear das k palavras-código da base. Para ilustrar este conceito, considere o código de taxa $R = 2/3$ do Exemplo 47. Se escolhermos as palavras-código $(1, 0, 1)$ e $(0, 1, 1)$ para formar a base, e considerarmos todos os possíveis valores do vetor de informação $\mathbf{u} = (u_0, u_1)$, poderemos escrever qualquer palavra-código $\mathbf{v}$ na forma:

$$\mathbf{v} = (u_0 \times (1, 0, 1)) \oplus (u_1 \times (0, 1, 1))$$

Por exemplo, para $\mathbf{u} = (u_0, u_1) = (1, 1)$, temos a palavra-código (1,1,0).

Podemos expressar a equação acima na forma matricial. Definindo a matriz

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

escrevemos

$$\begin{aligned} \mathbf{v} &= \mathbf{u}\,\mathbf{G} \\ (v_0, v_1, v_2) &= (u_0, u_1) \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \end{aligned}$$

Assim, para $\mathbf{u} = (1, 1)$, teremos

$$\mathbf{v} = (1, 1) \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} = (1, 1, 0)$$

como esperávamos.

A matriz $\mathbf{G}$ é chamada a *matriz geradora* de um código de bloco. Apenas os códigos de bloco lineares possuem matriz geradora. E ela não é única. Por exemplo, se trocarmos uma linha pela outra na matriz $\mathbf{G}$ do exemplo acima ainda geraremos o mesmo código, porém os vetores de informação serão associados às palavras-código numa ordem diferente. Considerando a Tabela 4.1, por exemplo, seria como se mantivéssemos a coluna $\mathbf{u}$ inalterada e permutássemos algumas das palavras-código na coluna $\mathbf{v}$. Como o código C é o conjunto formado por todas as palavras-código, e não tem nenhuma

relação com a ordem em que estas palavras-código são listadas, um matriz geradora equivalente, obtida por exemplo permutando-se linhas de $\mathbf{G}$, gera exatamente o mesmo código, apesar de modificar a relação entre $\mathbf{u}$ e $\mathbf{v}$.

Considerando agora o código de Hamming da Tabela 4.1, teremos a seguinte matriz geradora:

$$\begin{aligned} \mathbf{G} &= \begin{bmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{4.6}$$

Assim, se $\mathbf{u} = (1, 1, 0, 1)$, teremos

$$\begin{aligned} \mathbf{v} &= \mathbf{u}\,\mathbf{G} \\ &= (1,1,0,1) \begin{bmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{bmatrix} \\ &= g_0 \oplus g_1 \oplus g_3 \\ &= (0,0,0,1,1,0,1) \end{aligned}$$

como podemos verificar na Tabela 4.1.

Em geral, para k e n quaisquer, teremos:

$$\mathbf{v} = \mathbf{u}\,\mathbf{G}$$

onde

$$\mathbf{u} = (u_0, u_1, \ldots, u_{k-1})$$

$$\mathbf{v} = (v_0, v_1, \ldots, v_{n-1})$$

e

$$\mathbf{G} = \begin{bmatrix} g_0 \\ g_1 \\ g_2 \\ \vdots \\ g_{k-1} \end{bmatrix} = \begin{bmatrix} g_{00} & g_{01} & g_{02} & \cdots & g_{0,n-1} \\ g_{10} & g_{11} & g_{12} & \cdots & g_{1,n-1} \\ g_{20} & g_{21} & g_{22} & \cdots & g_{2,n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ g_{k-1,0} & g_{k-1,1} & g_{k-1,2} & \cdots & g_{k-1,n-1} \end{bmatrix} \tag{4.7}$$

Em certas situações pode ser vantajoso ter os bits de informação inalterados na palavra-código. Um código de bloco linear é dito ser *sistemático* se os k bits de informação aparecem inalterados na palavra-código, ou seja, a palavra-código é escrita como:

$$\mathbf{v} = (v_0, v_1, \cdots, v_{n-k-1}, u_0, u_1, \cdots, u_{k-1})$$

A matriz geradora de um código sistemático é dada por

$$\mathbf{G} = \left[\mathbf{P}_{k\times(n-k)} : \mathbf{I}_k\right]$$

onde $\mathbf{I}_k$ é a matriz identidade de ordem k. Escrevendo esta matriz na forma expandida, temos:

$$\mathbf{G} = \begin{bmatrix} p_{00} & p_{01} & \cdots & p_{0,n-k-1} & 1 & & \\ p_{10} & p_{11} & \cdots & p_{1,n-k-1} & & 1 & \\ \vdots & \vdots & & \vdots & & & \ddots \\ p_{k-1,0} & p_{k-1,1} & \cdots & p_{k-1,,n-k-1} & & & 1 \end{bmatrix} \tag{4.8}$$

onde os espaços vazios na matriz denotam zeros.

Devemos notar que o código de Hamming de taxa 4/7 apresentado na Tabela 4.1 é sistemático. Observe que as suas palavras-código são obtidas por $\mathbf{v} = \mathbf{u}\ \mathbf{G} = [u_0, u_1, u_2, u_3]\ \mathbf{G}$. Ou seja,

$$\begin{aligned} v_0 &= u_0 + u_2 + u_3 \\ v_1 &= u_0 + u_1 + u_2 \\ v_2 &= u_1 + u_2 + u_3 \\ v_3 &= u_0 \\ v_4 &= u_1 \\ v_5 &= u_2 \\ v_6 &= u_3 \end{aligned}$$

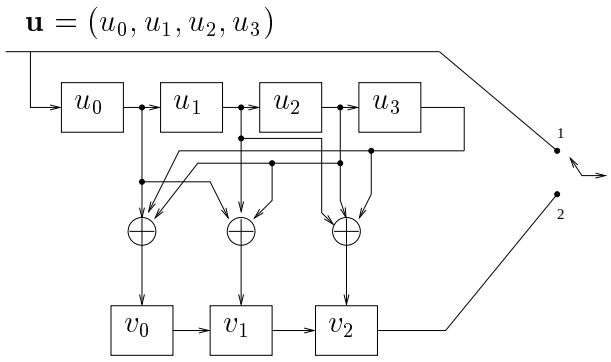


Figura 4.3: Codificador para o Código de Hamming Sistemático de Taxa 4/7.

A partir das equações acima, podemos facilmente implementar um circuito lógico para desempenhar o papel do codificador de Hamming. A Figura 4.3 mostra tal codificador.

4.1.4 A Matriz de Verificação de Paridade

Uma outra matriz associada a todo código de bloco linear é a *matriz de verificação de paridade*, denotada por $\mathbf{H}$. Seja C o espaço (conjunto de n-uplas) formado por todas as palavras-código de um código linear de taxa $R = k/n$ com matriz geradora $\mathbf{G}$. Da Teoria de Espaços Lineares, existe uma matriz $\mathbf{H}$ de dimensões $(n-k) \times n$ cujas linhas são linearmente independentes tal que qualquer vetor em C é ortogonal às linhas de $\mathbf{H}$, e que qualquer vetor binário de comprimento n que é ortogonal às linhas de $\mathbf{H}$ pertence a C. Desta maneira, este código C pode também ser descrito com base na matriz $\mathbf{H}$ da seguinte maneira:

> Uma n-upla $\mathbf{v}$ é uma palavra-código do código C gerado por $\mathbf{G}$ se e somente se
>
> $$\mathbf{v}\,\mathbf{H}^T = \mathbf{0}$$
>
> onde $\mathbf{H}$ é a matriz de verificação de paridade do código.

A Figura 4.4 ilustra o efeito das matrizes $\mathbf{G}$ e $\mathbf{H}$ de um código de bloco linear.

O nome da matriz $\mathbf{H}$ (verificação de paridade) segue do fato de que esta matriz pode ser usada para *verificar* se uma n-upla binária é ou não uma palavra-código.

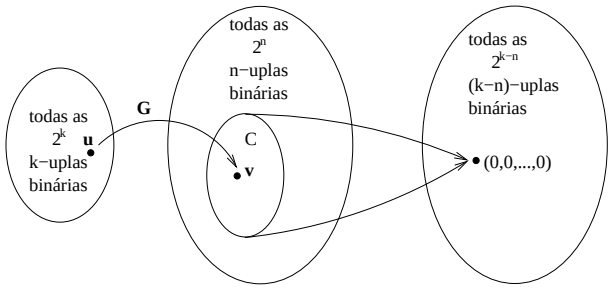


Figura 4.4: Diagram ilustrando o efeito das matrizes **G** e **H** de um código de bloco linear.

No caso de o código de bloco linear ser sistemático, a matriz de verificação de paridade **H** terá uma forma bastante simples. Considere a matriz geradora em (4.8). Na forma sistemática, temos que:

$$\begin{aligned}\mathbf{H} &= [\mathbf{I}_{n-k} \;\; \mathbf{P}^T] \qquad (4.9)\\ &= \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & p_{00} & p_{10} & \cdots & p_{k-1,0} \\ 0 & 1 & 0 & \cdots & 0 & p_{01} & p_{11} & \cdots & p_{k-1,1} \\ 0 & 0 & 1 & \cdots & 0 & p_{02} & p_{12} & \cdots & p_{k-1,2} \\ & & & & & . & & & \\ & & & & & . & & & \\ & & & & & . & & & \\ 0 & 0 & 0 & \cdots & 1 & p_{0,n-k-1} & p_{1,n-k-1} & \cdots & p_{k-1,n-k-1} \end{bmatrix}\end{aligned}$$

Exemplo 49 *Considere o código de Hamming de taxa 4/7 apresentado na Tabela 4.1. A partir de (4.6), reconhecemos que a matriz* **P** *é dada por:*

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

Logo, a matriz de verificação de paridade para o código da Tabela 4.1 é dada por:

$$\begin{aligned}\mathbf{H} &= [\mathbf{I}_3 \;\; \mathbf{P}^T] \\ &= \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}\end{aligned}$$

Tomemos como exemplo a palavra-código $\mathbf{v} = (1,1,0,0,1,0,1)$. *Podemos verificar que:*

$$\begin{aligned}
\mathbf{v}\,\mathbf{H}^T &= (1,1,0,0,1,0,1)\begin{bmatrix} 1&0&0&1&0&1&1\\ 0&1&0&1&1&1&0\\ 0&0&1&0&1&1&1 \end{bmatrix}^T \\
&= (1,1,0,0,1,0,1)\begin{bmatrix} 1&0&0\\ 0&1&0\\ 0&0&1\\ 1&1&0\\ 0&1&1\\ 1&1&1\\ 1&0&1 \end{bmatrix} \\
&= (0,0,0)
\end{aligned}$$

Exemplo 50 *Consideremos a mesma palavra-código do exemplo anterior, mas agora supondo que o vetor erro tenha sido* $\mathbf{e} = (0,0,0,0,0,0,1)$. *Ou seja, teremos o vetor recebido* $\mathbf{r} = (1,1,0,0,1,0,0)$. *Podemos verificar que:*

$$\begin{aligned}
\mathbf{r}\,\mathbf{H}^T &= (1,1,0,0,1,0,0)\begin{bmatrix} 1&0&0\\ 0&1&0\\ 0&0&1\\ 1&1&0\\ 0&1&1\\ 1&1&1\\ 1&0&1 \end{bmatrix} \\
&= (1,0,1)
\end{aligned}$$

Como sabíamos, $\mathbf{r}$ *não é uma palavra-código.*

É importante observar que a matriz de paridade $\mathbf{H}$ tem $k' = n - k$ linhas linearmente independentes, e que cada linha de $\mathbf{H}$ tem comprimento n. Pela definição de matriz geradora de um código linear, podemos concluir que $\mathbf{H}$ é a matriz geradora de um código linear de taxa $R' = k'/n$. Seja C o código gerado pela matriz geradora $\mathbf{G}$ e que tem como matriz de verificação de paridade a matriz $\mathbf{H}$. Então o código C_d,

chamado de *código dual* de C, é aquele código cuja matriz geradora é $\mathbf{H}$ e que tem como matriz de verificação de paridade a matriz $\mathbf{G}$. Devemos notar que, para qualquer $\mathbf{v} \in C$ e qualquer $\mathbf{w} \in C_d$, temos que $\mathbf{v}\,\mathbf{w}^T = 0$. Vejamos um exemplo.

Exemplo 51 *Considere o código (sistemático) de repetição de taxa $R = k/n = 1/3$. A matriz geradora desse código é $\mathbf{G} = [\mathbf{P}\ \mathbf{I}_1] = [1, 1, 1]$ e a matriz de verificação de paridade é*

$$\begin{aligned} \mathbf{H} &= [I_2\ \mathbf{P}^T] \\ &= \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \end{aligned}$$

O código dual do código de repetição de taxa $R = 1/3$ é o código de taxa $R' = (n - k)/n = 2/3$ geradado pela matriz geradora

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

que é o código de verificação de paridade de taxa 2/3 apresentado no Exemplo 47. Note que as palavras-código do código dual são: (0,0,0), (0,1,1), (1,0,1) e (1,1,0), obtidas através de combinações lineares da matriz geradora.

Estendendo o resultado deste exemplo, dizemos que o código dual do código (sistemático) de repetição de taxa $R = 1/n$ é o código de verificação de paridade de taxa $R' = (n - 1)/n$.

4.1.5 A Distância de Hamming Mínima de um Código de Bloco

Nesta seção, definiremos o importante conceito de distância de Hamming mínima de um código de bloco. Como veremos mais adiante, este é o parâmetro mais importante de um código, e aquele que determina a sua capacidade de detecção e de correção. A determinação deste parâmetro é bastante simplificada quando o código em questão é linear. Através de uma análise das colunas da matriz de verificação de paridade $\mathbf{H}$, pode-se facilmente determinar a distância de Hamming mínima de um código linear.

Seja $d_H(\mathbf{v}, \mathbf{v}')$ a distância de Hamming entre as n-uplas $\mathbf{v}$ e $\mathbf{v}'$, definida na Seção 4.1.1. Vamos também considerar o *peso de Hamming* de uma n-upla $\mathbf{v}$, ou seja, o número de elementos (ou coordenadas) de $\mathbf{v}$ que são diferentes de zero. Denotemos por $w_H(\mathbf{v})$ o peso de Hamming de $\mathbf{v}$. Por exemplo, dado que $\mathbf{v} = (1, 1, 0)$ temos que $w_H(\mathbf{v}) = 2$. É facil verificar que, para quaisquer n-uplas binárias $\mathbf{v}$ e $\mathbf{v}'$, temos que:

$$d_H(\mathbf{v}, \mathbf{v}') = w_H(\mathbf{v} \oplus \mathbf{v}') \tag{4.10}$$

Exemplo 52 *Consideremos as duas n-uplas $\mathbf{v} = (1, 1, 0, 1, 0, 0, 1)$ e $\mathbf{v}' = (1, 0, 0, 0, 1, 1, 0)$. A distância de Hamming é $d_H(\mathbf{v}, \mathbf{v}') = 5$, e o peso de Hamming da soma é*

$$w_H((1, 1, 0, 1, 0, 0, 1) \oplus (1, 0, 0, 0, 1, 1, 0)) = w_H((0, 1, 0, 1, 1, 1, 1)) = 5$$

Logo, $d_H(\mathbf{v}, \mathbf{v}') = w_H(\mathbf{v} \oplus \mathbf{v}')$.

A igualdade em (4.10) se verifica pois justamente nas posições em que $\mathbf{v}$ e $\mathbf{v}'$ diferem o resultado da soma módulo 2 é 1. Assim o número de 1's em $\mathbf{v} \oplus \mathbf{v}'$, ou seja $w_H(\mathbf{v})$, é justamente o número de posições em que $\mathbf{v}$ e $\mathbf{v}'$ diferem, ou seja, $d_H(\mathbf{v}, \mathbf{v}')$.

A *distância de Hamming mínima* de um código de bloco linear C, denotada por $d_{\min}(C)$, ou simplesmente $d_{\min}$, é definida da seguinte maneira:

$$d_{\min} \stackrel{\Delta}{=} \min \{d_H(\mathbf{v}, \mathbf{v}') | \mathbf{v} \in C, \mathbf{v}' \in C, \mathbf{v} \neq \mathbf{v}'\} \tag{4.11}$$

Exemplo 53 *Considere o código de bloco linear de taxa $R = 2/3$ apresentado no Exemplo 47, cujas palavras-código são: $\mathbf{v}_1 = (0, 0, 0), \mathbf{v}_2 = (0, 1, 1), \mathbf{v}_3 = (1, 0, 1)$ e $\mathbf{v}_4 = (1, 1, 0)$. Formando-se todos os 6 possíveis pares de palavras-código, quais sejam,*

$$(\mathbf{v}_1, \mathbf{v}_2), (\mathbf{v}_1, \mathbf{v}_3), (\mathbf{v}_1, \mathbf{v}_4), (\mathbf{v}_2, \mathbf{v}_3), (\mathbf{v}_2, \mathbf{v}_4) \quad e \quad (\mathbf{v}_3, \mathbf{v}_4)$$

tem-se que as respectivas distâncias de Hamming são, 2, 2, 2, 2, 2 e 2. Logo, a distância mínima deste código é $d_{\min} = 2$.

Vamos agora supor que o código em questão seja linear. Neste caso, temos que

$$\begin{aligned} d_{\min} &= \min \{d_H(\mathbf{v}, \mathbf{v}') | \mathbf{v} \in C, \mathbf{v}' \in C, \mathbf{v} \neq \mathbf{v}'\} && (4.12) \\ &= \min \{w_H(\mathbf{v}'') | \mathbf{v}'' \in C, \mathbf{v}'' \neq \mathbf{0}\} && (4.13) \end{aligned}$$

onde a segunda igualdade segue do fato de que a soma de duas palavras-código é uma palavra-código se o código for linear. Assim, para um código de bloco linear, a distância de Hamming mínima é igual ao peso de Hamming mínimo de suas palavras-código não nulas.

Exemplo 54 *No exemplo anterior, bastaria ter verificado o peso de Hamming de todas as palavras-código não nulas, quais sejam, (0,1,1), (1,0,1) e (1,1,0). Como os pesos de Hamming são 2, 2 e 2, o peso de Hamming mínimo (ou distância de Hamming mínima) do código é 2.*

A vantagem conseguida em relação ao cálculo da distância mínima de um código linear sobre um código não linear é ainda mais evidente quando o número de palavras-código é grande.

Exemplo 55 *Considere o código de Hamming de taxa 4/7 apresentado na Tabela 4.1. Como são 16 palavras-código, há* $\binom{16}{2} = 120$ *pares distintos de palavras-código. Portanto, seriam necessários 120 cálculos de distância para se chegar à distância mínima. Devido à linearidade deste código, podemos verificar facilmente que o peso mínimo de uma palavra-código não nula (há apenas 15 delas) na Tabela 4.1 é 3. Logo este código tem* $w_{\min} = d_{\min} = 3$.

4.1.6 As capacidades de Detecção e de Correção de um Código Linear

Dado um código de bloco de taxa $R = k/n$ e cuja distância mínima é $d_{\min}$, até quantos erros de bit ele é capaz de detectar? E quanto a correção de erros, qual a capacidade do código? Qual a razão para se dizer que o código de taxa 2/3 do Exemplo 47 é capaz de detectar um único erro, e não tem capacidade para corrigir nenhum erro? Por outro lado, o que nos permite afirmar que o código de Hamming da Tabela 4.1 é capaz de corrigir um erro de bit? Nesta seção iremos responder essas perguntas, com justificativas.

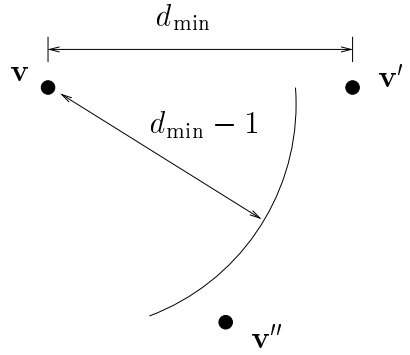


Figura 4.5: Espaço contendo três palavras-código de um código de bloco.

A Capacidade de Detecção

Dizemos que um código tem *capacidade de detecção* de N_D erros de bit se ele for capaz de detectar qualquer padrão de erro $\mathbf{e}$ tal que $w_H(\mathbf{e}) \leq N_D$, e houver pelo menos um padrão de erro $\mathbf{e}$ com $w_H(\mathbf{e}) = N_D + 1$ tal que o código não seja capaz de detectar. É importante observar que detectamos a presença de erro no canal quando o vetor recebido $\mathbf{r}$ não for uma palavra-código, pois apenas palavras-código são transmitidas através do canal. Porém, há a possibilidade de o vetor recebido $\mathbf{r}$ ser uma palavra-código, mas não ser a palavra-código transmitida. Isso acontece se e somente se o vetor erro $\mathbf{e}$ for uma palavra-código não nula. Note que se o vetor erro $\mathbf{e}$ for uma palavra-código não nula, então teremos que $\mathbf{r} = \mathbf{v} \oplus \mathbf{e} \neq \mathbf{v}$ também será uma palavra-código, pois o código é linear. Assim, podemos afirmar que os erros não detectáveis são aqueles vetores $\mathbf{e}$ que pertencem ao código, ou seja, que são palavras-código.

Para entender melhor o conceito de capacidade de detecção de um código de bloco, devemos imaginar as palavras-código do código como pontos em um espaço de dimensão n, semelhante ao espaço mostrado na Figura 4.2. Consideremos duas palavras-código $\mathbf{v}$ e $\mathbf{v}'$, distantes uma da outra de $d_H(\mathbf{v}, \mathbf{v}') = d_{\min}$, e uma terceira palavra-código $\mathbf{v}''$, mais afastada, como pontos no espaço, como indicado na Figura 4.5. Devemos notar que, se o vetor erro $\mathbf{e}$ tiver peso de Hamming $w_H(\mathbf{e}) \leq d_{\min} - 1$, então a palavra-código transmitida $\mathbf{v}$ estará a uma distância menor ou igual a $d_{\min} - 1$ do vetor recebido $\mathbf{r} = \mathbf{v} \oplus \mathbf{e}$. Isto significa que $\mathbf{r}$ não pode ser uma palavra-código. Portanto, diremos com certeza que o código é capaz de detectar todos os padrões de erro cujo peso de Hamming seja $\leq d_{\min} - 1$. Por outro lado, vamos supor que o vetor erro $\mathbf{e}$ seja uma palavra-código de peso exatamente $d_{\min}$. Como o código tem distância

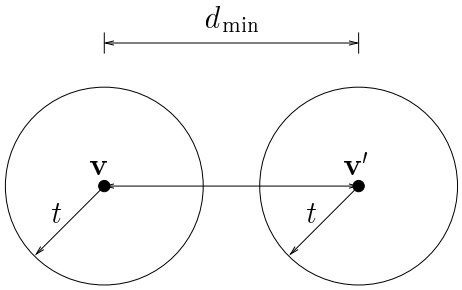


Figura 4.6: Espaço ilustrando a capacidade de correção de erro de um código de bloco.

mínima (ou peso mínimo) $d_{\min}$, então existe pelo menos uma palavra-código de peso exatamente $d_{\min}$. Neste caso, o vetor recebido seria uma palavra-código, e o código não seira capaz de detectar os $d_{\min}$ erros de bit ocorridos no canal. Concluimos portanto que:

A capacidade de detecção de um código de bloco é $d_{\min} - 1$.

Por exemplo, para o código de taxa 2/3 do Exemplo 47, como $d_{\min} = 2$ temos que o código detecta até $d_{\min} - 1 = 1$ erro de bit.

A Capacidade de Correção

Dizemos que um código tem *capacidade de correção* de N_C erros de bit se ele for capaz de corrigir qualquer padrão de erro $\mathbf{e}$ tal que $w_H(\mathbf{e}) \leq N_C$, e houver pelo menos um padrão de erro $\mathbf{e}$ com $w_H(\mathbf{e}) = N_C + 1$ tal que o código não seja capaz de corrigir.

Diferente do caso da detecção, no processo de correção o decodificador tentará corrigir o(s) erro(s) quando o vetor recebido $\mathbf{r}$ não for uma palavra-código. Porém, se o número de erros ultrapassar a capacidade de correção do código, haverá uma decodificação errônea. Estes são chamados de os padrões de erro não corrigíveis.

Para entender melhor o conceito de capacidade de correção de um código de bloco, vamos considerar a Figura 4.6, onde são mostradas duas palavras-código $\mathbf{v}$ e $\mathbf{v}'$, distantes uma da outra de $d_H(\mathbf{v}, \mathbf{v}') = d_{\min}$. Vamos agora imaginar 2^k esferas de raio t soltas no espaço, com centros exatamente nos pontos correspondentes às palavras-código. Vamos supor que a palavra-código $\mathbf{v}$, que é o centro da esfera $E_{\mathbf{v}}$, tenha sido transmitida. O decodificador usará a seguinte regra de decodificação:

1. Se o vetor recebido $\mathbf{r}$ estiver dentro ou na superfície da esfera $E_{\mathbf{v}}$, então a palavra-código correspondendo ao centro desta esfera (ou seja, $\mathbf{v}$) será a palavra decodificada (dada como correta). Com esta regra, se o número de erros for $\leq t$, o decodificador será capaz de corrigir o(s) erro(s) com sucesso.

2. Se o vetor recebido $\mathbf{r}$ estiver dentro de uma outra esfera, digamos $E_{\mathbf{v}'}$, então a palavra decodificada será $\mathbf{v}'$, e a decodificação será errônea.

3. Se o vetor recebido $\mathbf{r}$ estiver fora de qualquer esfera, o decodificador poderá decidir-se por uma das palavras-código mais próximas, ou declarar apenas que houve erro na transmissão.

Devemos notar que as esferas não podem se tocar, pois se isso acontecer, e o vetor recebido estiver neste ponto de toque, o decodificador ficaria na indecisão. Por outro lado, quanto maior for o raio t das esferas, maior será o número de erros corrigíveis. A partir destas duas imposições, devemos ter a situação indicada na Figura 4.6, onde a distância entre as supefícies de duas esferas vizinhas deve ser pelo menos de 1 bit. Em outras palavras,

$$d_{\min} \geq 2t + 1$$

Já podemos notar que o número de erros que o código pode corrigir é até t erros de bit, e ficamos com o seguinte para a capacidade de correção de um código de bloco:

A capacidade de correção de um código de bloco é

$$t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$$

onde $\lfloor x \rfloor$ é o maior inteiro menor ou igual a x.

Por exemplo, para o código de Hamming da Tabela 4.1, como $d_{\min} = 3$ temos que

$$t = \left\lfloor \frac{3 - 1}{2} \right\rfloor = 1$$

Note que, para o código de taxa 2/3 do Exemplo 47, temos $t = 0$, ou seja, aquele código não é um código corretor de erro, é apenas um código detector de erro.

4.1.7 Determinação da Distância Mínima via a Matriz H

Como vimos anteriormente, o prodecimento para a determinação da distância mínima de um código de bloco é bem mais simples se o código de bloco for linear. Nesta seção, mostraremos que a propriedade da linearidade pode simplificar ainda mais a determinação da distância mínima.

Vamos supor que a palavra-código $\mathbf{v} = (v_0, v_1, \ldots, v_{n-1})$ tenha peso de Hamming l, e que as l posições iguais a 1 nesta palavra sejam: $i_1, i_2, \ldots, i_l$, onde $0 \leq i_1 < i_2 < \ldots < i_l \leq n-1$. Seja

$$\mathbf{H} = [\mathbf{h}_0, \mathbf{h}_1, \ldots, \mathbf{h}_{n-1}]$$

a matriz de verificação de paridade do código, onde $\mathbf{h}_i$ é a i-ésima coluna de $\mathbf{H}$. Como $\mathbf{v}$ é uma palavra-código, então devemos ter:

$$\begin{aligned}
\mathbf{v}\,\mathbf{H}^T &= 0 \\
(v_0, v_1, \ldots, v_{n-1}) \begin{bmatrix} \mathbf{h}_0^T \\ \mathbf{h}_1^T \\ \vdots \\ \mathbf{h}_{n-1}^T \end{bmatrix} &= [0, 0, \cdots, 0] \\
v_0\mathbf{h}_0^T + v_1\mathbf{h}_1^T + \cdots + v_{n-1}\mathbf{h}_{n-1}^T &= [0, 0, \cdots, 0] \\
v_{i_1}\mathbf{h}_{i_1}^T + v_{i_2}\mathbf{h}_{i_2}^T + \cdots + v_{i_l}\mathbf{h}_{i_l}^T &= [0, 0, \cdots, 0] \\
\mathbf{h}_{i_1}^T + \mathbf{h}_{i_2}^T + \cdots + \mathbf{h}_{i_l}^T &= [0, 0, \cdots, 0]
\end{aligned}$$

A partir da última equação acima podemos concluir que para cada $\mathbf{v} \in C$ de peso de Hamming l, existem l colunas de $\mathbf{H}$ cuja soma é zero, ou seja, existem l colunas de $\mathbf{H}$ que são linearmente dependentes. Também é fácil verificar que para cada l colunas de $\mathbf{H}$ cuja soma é zero (linearmente dependentes), existe uma palavra-código $\mathbf{v}$ de peso de Hamming l cujas posições em que um 1 ocorrem coincidem com as posições destas l colunas de $\mathbf{H}$. A partir desta propriedade, temos o seguinte resultado.

Teorema 9 *Seja C um código linear com matriz de verificação de paridade $\mathbf{H}$. O peso mínimo (logo a distância mínima) de C é igual ao menor número de colunas de $\mathbf{H}$ cuja soma dá o vetor zero.*

Exemplo 56 *Considere o código de repetição de taxa 1/3, cujas palavras-código são: (0,0,0) e (1,1,1). A matriz geradora deste código é* $\mathbf{G} = [1, 1, 1] = [\mathbf{P}\ \mathbf{I}_k]$. *A matriz de verificação de paridade é*

$$\mathbf{H} = [\mathbf{I}_{n-k}\ \mathbf{P}^T] = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Como não há 2 colunas de $\mathbf{H}$ *cuja soma é zero, e as 3 colunas de* $\mathbf{H}$ *somam zero, temos que* $d_{\min} = 3$.

Exemplo 57 *Considere agora o código de paridade de taxa 2/3 cuja matriz geradora é*

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

e cuja matriz de verificação de paridade é $\mathbf{H} = [1\ 1\ 1]$. *Como a soma de duas colunas quaisquer é zero, a distâcia mínima deste código é* $d_{\min} = 2$.

Exemplo 58 *Considere agora o código de Hamming de taxa 4/7. A matriz de verificação de paridade é*

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

Note que todas as colunas são não nulas, logo $d_{\min} > 1$. *Também, não há duas colunas idênticas (ou seja, cuja soma seja zero), logo* $d_{\min} > 2$. *Porém, se escolhermos por exemplo as colunas 1, 2 e 4, teremos três colunas cuja soma é zero, logo* $d_{\min} = 3$.

Com base na discussão acima sobre a determinação da distância mínima de um código de bloco linear a partir da análise das colunas da matriz $\mathbf{H}$, podemos agora nos voltar para o problema de construção de códigos ótimos, ou seja, códigos com a maior $d_{\min}$ possível. Note que devemos considerar a matriz $\mathbf{H}$ neste processo de construção. Também podemos nos restringir às matrizes $\mathbf{H}$ na forma sistemática; um código de bloco não sistemático não pode ser melhor que um sistemático, visto que qualquer matriz geradora na forma não sistemática poderá, através de operações de linha, ser convertida numa forma sistemática, e esta última gerará exatamente o mesmo código.

Uma vez encontrada a matriz $\mathbf{H}$ na forma sistemática de um código com a maior distância mínima possível, podemos facilmente encontrar a matriz geradora $\mathbf{G}$ deste código, como mostrado na equação (4.9). Vejamos um exemplo.

Vamos construir um código de bloco linear e sistemático ótimo de taxa $R = k/n = 3/6$. A matriz de verificação de paridade na forma sistemática é da forma:

$$\mathbf{H} = [\mathbf{I}_{n-k}\ \mathbf{P}^T] = \begin{bmatrix} 1 & 0 & 0 & p_{00} & p_{10} & p_{20} \\ 0 & 1 & 0 & p_{01} & p_{11} & p_{21} \\ 0 & 0 & 1 & p_{02} & p_{12} & p_{22} \end{bmatrix}$$

Note que temos 9 incógnitas, que correspondem aos elementos da matriz $\mathbf{P}$. Portanto, há $2^9 = 512$ possíveis códigos de bloco lineares e sistemáticos de taxa 3/6. As 3 primeiras colunas de $\mathbf{H}$ já foram escolhidas (a matriz $\mathbf{I}_{n-k}$). Devemos escolher as 3 colunas restantes. Como cada coluna tem 3 bits, temos 8 possibilidades para essas colunas. Obviamente, dentre essas 8 possibilidades devemos desconsiderar a coluna zero, pois caso contrário teríamos **uma** coluna "cuja soma" é zero, e ficaríamos com $d_{\min} = 1$. Também devemos eliminar as colunas que já foram utilizadas (ou seja, as colunas da matriz $\mathbf{I}_{n-k}$, caso contrário, com duas colunas idênticas, ficaríamos restritos a $d_{\min} = 2$. Assim, devemos preencher as 3 colunas da direita de $\mathbf{H}$ a partir da seguinte lista de colunas:

$$\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \quad \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \quad \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \quad \text{e} \quad \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

Agora devemos notar que como cada coluna de $\mathbf{H}$ é um vetor tridimensional, é impossível neste caso encontrarmos 4 colunas de $\mathbf{H}$ que sejam linearmente independentes. Logo, uma boa escolha seria:

$$\mathbf{H} = [\mathbf{I}_{n-k}\ \mathbf{P}^T] = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

e teremos $d_{\min} = 3$, pois somando-se as colunas 2, 3, e 6, por exemplo, obtemos a coluna zero.

Devemos observar que preenchendo a parte esquerda da matriz $\mathbf{H}$ com a matriz $\mathbf{I}_{n-k}$, estamos inicialmente selecionando $n - k$ colunas linearmente independentes de

$\mathbf{H}$, e com isso há a esperança de encontrarmos um código com $d_{\min} > n - k$. Mas não é possível incluirmos mais uma coluna e termos $n - k + 1$ colunas linearmente independentes, pois as colunas de $\mathbf{H}$ são vetores de dimensão $(n - k)$, e no espaço $(n - k)$-dimensional não há $n - k + 1$ vetores linearmente independentes. Assim, ficamos com o seguinte limitante superior para $d_{\min}$.

Teorema 10 (O limitante de Singleton) *Para um código de bloco linear de taxa $R = k/n$, temos que*

$$d_{\min} \leq n - k + 1$$

que é o limitante de Singleton.

No exemplo anterior, temos que $d_{\min} \leq 6 - 3 + 1 = 4$. O melhor código de bloco de taxa $3/6$ tem $d_{\min} = 3 < 4$.

4.1.8 Detecção e Correção de Erros: Os Conceitos de Síndrome e Arranjo Padrão

A *síndrome* de um vetor recebido $\mathbf{r}$ é definida como o vetor:

$$\mathbf{s} \stackrel{\Delta}{=} \mathbf{r}\,\mathbf{H}^T = [s_0, s_1, \ldots, s_{n-k-1}]$$

Obviamente,

Se $\mathbf{r} \in C$, então $\mathbf{s} = [0, 0, \ldots, 0]$

Se $\mathbf{r} \notin C$, então $\mathbf{s} \neq [0, 0, \ldots, 0]$

Vamos supor que a palavra-código $\mathbf{v}$ tenha sido transmitida e que o vetor erro tenha sido $\mathbf{e}$. Assim, o vetor recebido será

$$\mathbf{r} = \mathbf{v} \oplus \mathbf{e}$$

e a síndrome será dada por

$$\begin{aligned} \mathbf{s} &= \mathbf{r}\,\mathbf{H}^T \\ &= (\mathbf{v} \oplus \mathbf{e})\,\mathbf{H}^T \\ &= \mathbf{v}\,\mathbf{H}^T \oplus \mathbf{e}\,\mathbf{H}^T \\ &= \mathbf{e}\,\mathbf{H}^T \end{aligned}$$

onde usamos o fato de que $\mathbf{v}\ \mathbf{H}^T = \mathbf{0}$. Podemos notar que a síndrome não depende da palavra-código transmitida, depende apenas do vetor erro $\mathbf{e}$. Se $\mathbf{e} \neq \mathbf{0}$, ou seja, se houver erro, então a síndrome é diferente de $\mathbf{0}$. Portanto a síndrome pode ser usada para testar se o vetor recebido $\mathbf{r}$ é ou não uma palavra-código.

Vamos considerar o código de Hamming de taxa 4/7, cuja matriz de verificação de paridade é:

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

Seja $\mathbf{r} = (r_0, r_1, \ldots, r_6)$ o vetor recebido. Então, a síndrome $\mathbf{s} = (s_0, s_1, s_2)$ será:

$$\begin{aligned} \mathbf{s} &= (s_0, s_1, s_2) \\ &= \mathbf{r}\ \mathbf{H}^T \\ &= (r_0, r_1, \ldots, r_6) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \end{aligned}$$

Assim, obtemos o seguinte sistema de equações:

$$\begin{aligned} s_0 &= r_0 + r_3 + r_5 + r_6 \\ s_1 &= r_1 + r_3 + r_4 + r_5 \\ s_2 &= r_2 + r_4 + r_5 + r_6 \end{aligned}$$

Das equações, podemos facilmente construir um circuito de cálculo de síndrome e de detecção de erro para o código de Hamming de taxa 4/7, como mostrado na Figura 4.7.

A seguir, vamos considerar o problema de correção de erros. Seja C um código de bloco linear de taxa $R = k/n$ e sejam $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{2^k}$ as suas palavras-código. Sabemos que independentemente da palavra-código transmitida, devido ao ruído o vetor recebido $\mathbf{r}$ pode ser qualquer n-upla binária. Um esquema de decodificação (correção) de erro

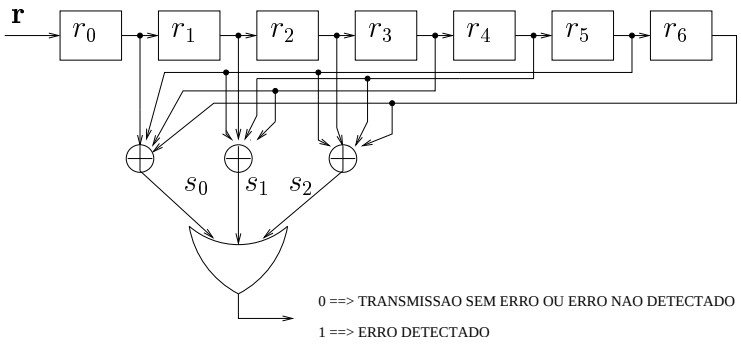


Figura 4.7: Circuito de cálculo de síndrome e de detecção de erros para o código de Hamming de taxa 4/7

é uma regra para particionar o conjunto de todos os 2^n possíveis vetores recebidos em 2^k subconuntos disjuntos $\mathbf{E}_1, \mathbf{E}_2, \ldots, \mathbf{E}_{2^k}$, tal que a palavra-código $\mathbf{v}_i$ e mais nenhuma outra palavra-código pertença ao subconjunto $\mathbf{E}_i$, para $1 \leq i \leq 2^k$. Assim, cada subconjunto estará associado a uma única palavra-código, e se o vetor recebido estiver no subconjunto $\mathbf{E}_i$, então a palavra-código decodificada (dada como correta) será $\mathbf{v}_i$.

Vamos considerar o seguinte método de partição do conjunto de todas as 2^n n-upla binárias em 2^k subconjuntos. Primeiramente, colocamos as 2^k palavras-código em uma linha de uma tabela, com o vetor $\mathbf{v}_1 = (0, 0, \ldots, 0)$ sendo o primeiro (mais à esquerda) elemento. Das $2^n - 2^k$ n-uplas restantes, escolhemos a n-upla $\mathbf{e}_2$ e a colocamos na segunda linha da tabela, logo abaixo do vetor $\mathbf{v}_1 = (0, 0, \ldots, 0)$. Completamos a segunda linha através da soma de $\mathbf{e}_2$ com a palavra-código imediatamente acima. Assim, teremos a n-upla $\mathbf{e}_2 + \mathbf{v}_i$ na segunda linha, imediatamente abaixo da palavra-código $\mathbf{v}_i$ da primeira linha da tabela. Tendo completado a segunda linha, escolhemos uma outra n-upla $\mathbf{e}_3$ que ainda não tenha sido utilizada na tabela, e a colocamos na primeira posição (mais à esquerda) na terceira linha. Procedemos como na segunda linha, ou seja, preenchendo a i-ésima posição da terceira linha com a soma $\mathbf{e}_3 + \mathbf{v}_i$. E assim por diante, até esgotarmos todas as n-uplas binárias. Esta tabela é chamada de

arranjo padrão, e é mostrada abaixo.

$$
\begin{array}{ccccccc}
\mathbf{v}_1 = \mathbf{0} & \mathbf{v}_2 & \cdots & \mathbf{v}_i & \cdots & \mathbf{v}_{2^k} \\
\mathbf{e}_2 & \mathbf{e}_2 + \mathbf{v}_2 & \cdots & \mathbf{e}_2 + \mathbf{v}_i & \cdots & \mathbf{e}_2 + \mathbf{v}_{2^k} \\
\mathbf{e}_3 & \mathbf{e}_3 + \mathbf{v}_2 & \cdots & \mathbf{e}_3 + \mathbf{v}_i & \cdots & \mathbf{e}_3 + \mathbf{v}_{2^k} \\
\vdots & & & & & \vdots \\
\mathbf{e}_l & \mathbf{e}_l + \mathbf{v}_2 & \cdots & \mathbf{e}_l + \mathbf{v}_i & \cdots & \mathbf{e}_l + \mathbf{v}_{2^k} \\
\vdots & & & & & \vdots \\
\mathbf{e}_{2^{n-k}} & \mathbf{e}_{2^{n-k}} + \mathbf{v}_2 & \cdots & \mathbf{e}_{2^{n-k}} + \mathbf{v}_i & \cdots & \mathbf{e}_{2^{n-k}} + \mathbf{v}_{2^k}
\end{array}
$$

Note que a tabela terá exatamente 2^{n-k} linhas e 2^k colunas, totalizando $2^{n-k} \times 2^k = 2^n$ elementos, ou seja, todas as n-uplas binárias possíveis. É fácil verificar que cada n-upla binária aparece uma única vez na tabela, e nenhuma n-upla binária deixa de aparecer na tabela. Em Álgebra Abstrata, cada linha do arranjo padrão é chamada de *classe lateral* ou *coset*, e o elemento mais à esquerda (primeira coluna) é chamado de o *líder* do coset. É importante notar que todos os vetores da l-ésima linha têm a mesma síndrome, que é a síndrome do líder $\mathbf{e}_l$, dada por: $\mathbf{e}_l\,\mathbf{H}^T$. Assim, teremos 2^{n-k} síndromes distintas associadas aos 2^{n-k} líderes $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_{2^{n-k}}$. Vejamos um exemplo.

Considere o código de taxa 3/6 projetado no na seção anterior, cuja matriz geradora é:

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

O aranjo padrão desse código é mostrado na Tabela 4.2.

A construção de um arranjo padrão de um código de bloco linear é um passo fundamental para o processo de decodificação. Vamos supor, no exemplo acima, que o vetor recebido seja $\mathbf{r} = (1, 1, 1, 0, 0, 1)$, que situa-se na quarta linha e na quarta coluna. O resultado da decodificação deve ser então que o vetor erro foi $\mathbf{e} = (0, 0, 1, 0, 0, 0)$, que é o líder do coset referente à quarta linha, e a palavra-código decodificada é $\hat{\mathbf{v}} = (1, 1, 0, 0, 0, 1)$. Se o erro tiver sido de fato $\mathbf{e} = (0, 0, 1, 0, 0, 0)$, então a decodificação estará correta, e o único erro ocorrido no terceiro bit (da esquerda para a direita) terá sido corrigido. Naturalmente, para minimizar a probabilidade de erro, note que devemos escolher os líderes de coset como sendo os padrões de erro mais prováveis de

Tabela 4.2: Arranjo padrão para o código de taxa 3/6.

líderes							
000000	011100	101010	110001	110110	101101	011011	000111
100000	111100	001010	010001	010110	001101	111011	100111
010000	001100	111010	100001	100110	111101	001011	010111
001000	010100	100010	111001	111110	100101	010011	001111
000100	011000	101110	110101	110010	101001	011111	000011
000010	011110	101000	110011	110100	101111	011001	000101
000001	011101	101011	110000	110111	101100	011010	000110
100100	111000	001110	010101	010010	001001	111111	100011

aconteceremem na transmissão. Estaremos assim corrigindo os erros que ocorrem mais freqüentemente. Para n transmissões pelo canal BSC com probabilidade de transição $p < 1/2$, os erros mais prováveis são os de menor peso de Hamming, pois a probabilidade de haver w erros em n transmissões pelo canal BSC é igual a $p^w(1-p)^{n-w}$. Quanto menor o número de erros w, maior será a probabilidade. É por essa razão que, na Tabela 4.2, escolhemos como líderes o vetor todo zero, os 6 vetores de peso de Hamming 1, e um vetor de peso de Hamming 2. Esses serão os padrões de erro *corrigíveis*. Finalmente, o processo de decodificação, chamado de Decodificação por Síndrome, pode ser resumido como:

Decodificação por Síndrome

Step 1. Calcule a síndrome do vetor recebido $\mathbf{r}$, $\mathbf{r}\,\mathbf{H}^T$.

Step 2. Localize o líder de coset $\mathbf{e}_l$ cuja síndrome é igual a $\mathbf{r}\,\mathbf{H}^T$. Então $\mathbf{e}_l$ é tomado como o padrão de erro causado pelo canal.

Step 3. Decodifique o vetor recebido $\mathbf{r}$ pela palavra-código $\hat{\mathbf{v}} = \mathbf{r} \oplus \mathbf{e}_l$.

O algoritmo acima poderá ser automatizado da seguinte maneira. Vamos considerar mais uma vez o código de Hamming de taxa 4/7 mostrado na Tabela 4.1. Teremos $2^{n-k} = 2^3 = 8$ líderes de coset para escolher. Para minimizar a probabilidade de erro, escolheremos os líderes como sendo os padrões de erro mais prováveis, ou seja, o vetor

Tabela 4.3: Tabela de decodificação para o código de Hamming da Tabela 4.1.

Síndrome	Líderes de coset
(s_0, s_1, s_2)	$(e_0, e_1, e_2, e_3, e_4, e_5, e_6)$
(1,0,0)	(1,0,0,0,0,0,0)
(0,1,0)	(0,1,0,0,0,0,0)
(0,0,1)	(0,0,1,0,0,0,0)
(1,1,0)	(0,0,0,1,0,0,0)
(0,1,1)	(0,0,0,0,1,0,0)
(1,1,1)	(0,0,0,0,0,1,0)
(1,0,1)	(0,0,0,0,0,0,1)

todo zero e todos os 7 vetores (de comprimento 7) de peso de Hamming igual 1. Esses líderes de coset e as respectivas síndromes são encontradas na Tabela 4.3. Suponha que o vetor $\mathbf{v} = (1,0,0,1,0,1,1)$ seja a palavra-código transmitida e $\mathbf{r} = (1,0,0,1,1,1,1)$ seja o vetor recebido. Para decodificar $\mathbf{r}$, nós computamos a síndrome de $\mathbf{r}$,

$$\mathbf{s} = (1,0,0,1,1,1,1) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} = (0,1,1)$$

A partir da Tabela 4.3, temos que (0,1,1) é à síndrome do líder de coset $\mathbf{e} = (0,0,0,0,1,0,0)$. Logo, o resultado da decodificação será:

$$\begin{aligned} \hat{\mathbf{v}} &= \mathbf{r} \oplus \mathbf{e} \\ &= (1,0,0,1,1,1,1) \oplus (0,0,0,0,1,0,0) \\ &= (1,0,0,1,0,1,1) \end{aligned}$$

que de fato foi a palavra-código transmitida.

Devemos notar que se o padrão de erro não for um líder de coset, então o resultado da decodificação estará incorreto. Mas a probabalidade de isso ocorrer será

pequena se os líderes forem escolhidos como os padrões de menor peso de Hamming.

Um fato curioso ocorre com o código de taxa 3/6. Sabemos, da seção anterior, que este código, tendo $d_{\min} = 3$, corrige todos os padrões de erro de peso $t = 1$, e eventualmente algum (ou alguns) padrão (ou padrões) de peso maior que 1. De fato, a partir da Tabela 4.2, vimos que esta escolha de arranjo padrão permite corrigir 2 erros caso o padrão de erro seja especificamente $\mathbf{e} = (1, 0, 0, 1, 0, 0)$. Mas não garantimos a correção de todos os padrões de erro de peso 2.

Devemos notar que a mesma sorte não ocorre com o código de Hamming (veja Tabela 4.3). Pelo fato de o código de Hamming ser um código capaz de corrigir todos os padrões de erro de peso até t, e mais nenhum outro padrão de erro, ele é dito ser um código *perfeito*.

Finalmente, como a síndrome de $\mathbf{r} = \mathbf{v} + \mathbf{e}$ é igual à síndrome de $\mathbf{e}$, a partir da Tabela 4.3, e seguindo o algoritmo da Decodificação por Síndrome apresentado acima, podemos facilmente construir um circuito lógico para o decodificador para o código de Hamming de taxa 4/7, que é capaz de corrigir qualquer padrão de 1 erro de bit. Este codificador é mostrado na Figura 4.8.

4.2 Códigos Convolucionais

Na seção anterior vimos que em códigos de bloco, k bits de informação entram no codificador e este produz n bits codificados. A palavra-código $\mathbf{v}_k$, no instante k, depende apenas do vetor de informação $\mathbf{u}_k$, no instante k:

$$\mathbf{v}_k = \mathbf{u}_k \, \mathbf{G}$$

Em *códigos convolucionais*, temos que $\mathbf{v}_k$ depende do vetor de informação $\mathbf{u}_k$, no instante k, e também dos vetores de informação $\mathbf{u}_{k-j}$, $1 \leq j \leq m$, nos m instantes anteriores, onde $m \geq 1$ é a *memória* do código:

$$\begin{aligned} \mathbf{v}_k &= \mathbf{u}_k \, \mathbf{G}_0 + \mathbf{u}_{k-1} \, \mathbf{G}_1 + \cdots \mathbf{u}_{k-m} \, \mathbf{G}_m \\ &= \sum_{j=0}^{m} \mathbf{u}_{k-j} \, \mathbf{G}_j \end{aligned} \tag{4.14}$$

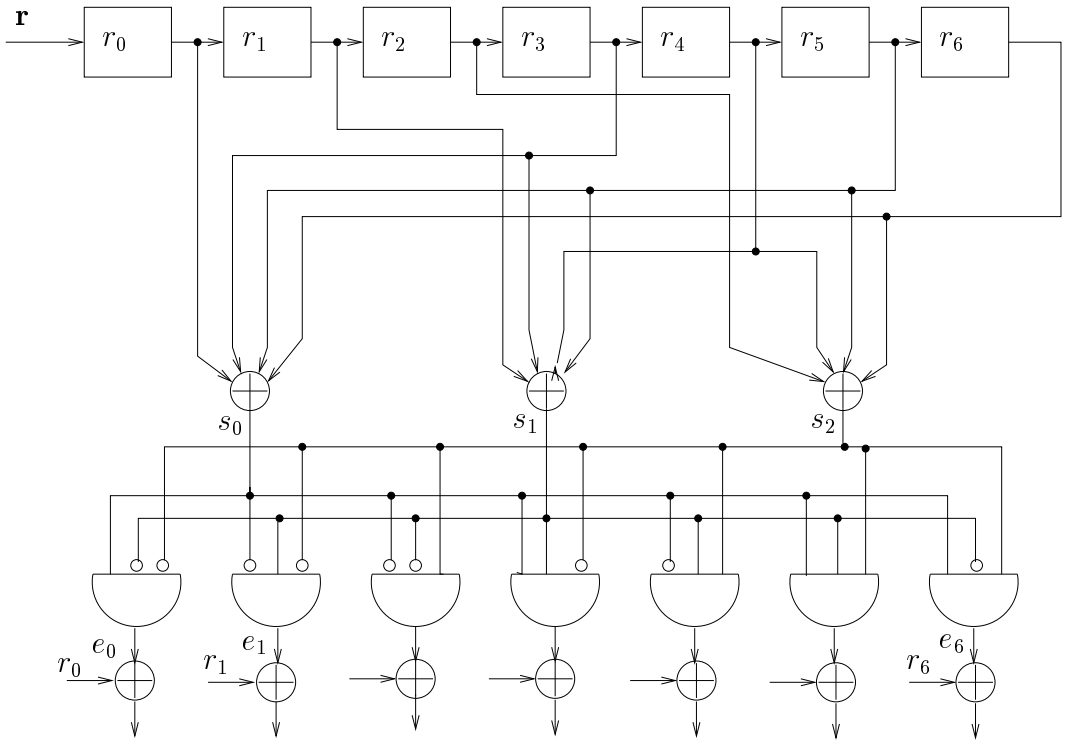


Figura 4.8: Circuito de decodificação para o código de Hamming de taxa 4/7.

onde $\mathbf{G}_j$, $0 \leq j \leq m$, são chamadas de *matrizes geradoras* do código convolucional de taxa $R = k/n$ e de memória m. Cada matriz $\mathbf{G}_j$ tem k linhas e n colunas.

Devemos notar que a equação acima é uma convolução discreta, daí o nome códigos "convolucionais".

Exemplo 59 *Vamos considerar um código convolucional de taxa $R = 1/2$ e de memória $m = 1$ descrito pelas seguintes matrizes geradoras:*

$$\mathbf{G}_0 = [1 \;\; 1] \;\; \mathbf{G}_1 = [0 \;\; 1]$$

Assim, de acordo com a equação (4.14), temos o correspondente codificador convolucional (circuito seqüencial) mostrado na Figura 4.9.

Como podemos ver, um codificador convolucional é uma máquina de estados finitos. Podemos definir o estado da máquina como o conteúdo mais atual do *flip-flop*. Como existe neste codificador um único *flip-flop* apenas, esta máquina contém $2^1 = 2$ estados (a saber, 0 e 1). Desta maneira, podemos representar este codificador por um diagrama de estados de 2 estados, como mostrado na Figura 4.10. Nesta figura, os

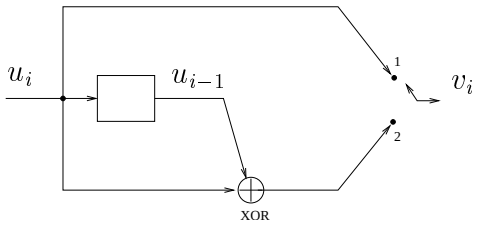


Figura 4.9: Codificador convolucional de taxa R=1/2 e memória $m = 1$ do Exemplo 59.

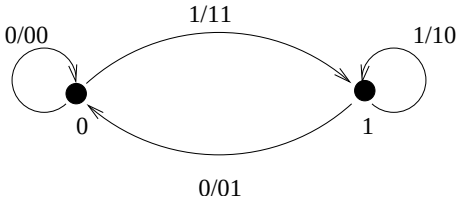


Figura 4.10: Diagrama de estados para o codificador convolucional de taxa R=1/2 e memória $m = 1$ do Exemplo 59.

rótulos a/bb denotam o bit de informação a e a correspondente palavra de ramo bb. Qualquer caminho neste diagrama que comece e termine no estado zero é dito ser uma *palavra-código* ou *seqüência-código* do código convolucional. Note que, diferente dos códigos de blocos, os código convolucionais considera seqüências codificadas, ao invés de blocos (de n bits) codificados. Uma palavra-código ou uma seqüência-código do código convolucional é portanto uma concatenação de vários blocos de n bits.

Podemos ainda representar as seqüências-código de um código convolucional através de uma treliça. A treliça pode ser entendida como uma expansão do diagrama de estados que indica a evolução temporal do codificador. A treliça para o código do Exemplo 59 é mostrada na Figura 4.11. Os rótulos sobre os ramos, nesta treliça, detonam o bit de informação e a correpondente palavra-ramo codificada, da mesma

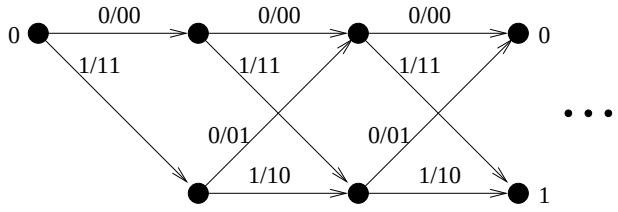


Figura 4.11: Treliça para o código convolucional de taxa R=1/2 e memória $m = 1$ do Exemplo 59.

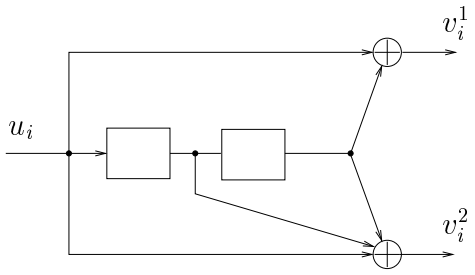


Figura 4.12: Codificador convolucional de taxa R=1/2 e memória $m = 2$ do Exemplo 60.

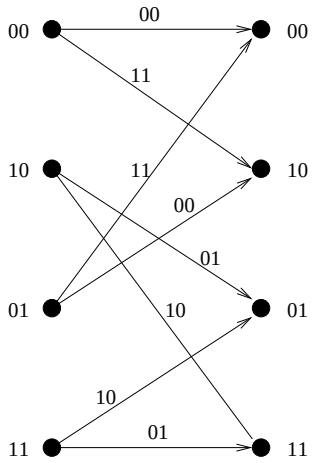


Figura 4.13: Treliça para o código convolucional de taxa R=1/2 e memória $m = 2$ do Exemplo 60.

maneira que na Figura 4.10. Vejamos um outro exemplo, agora com memória $m = 2$.

Exemplo 60 *Vamos considerar um código convolucional de taxa $R = 1/2$ e de memória $m = 2$ descrito pelas seguintes matrizes geradoras:*

$$\mathbf{G}_0 = [1 \;\; 1] \;\; \mathbf{G}_1 = [0 \;\; 1] \;\; \mathbf{G}_2 = [1 \;\; 1]$$

De acordo com a equação (4.14), temos o correspondente codificador convolucional (circuito seqüencial) mostrado na Figura 4.12.

Neste exemplo, como so existem dois *flip-flops*, a máquina contém $2^2 = 4$ estados (a saber, 00, 01, 10 e 11). Desta maneira, podemos representar este codificador por um diagrama de estados de 4 estados. A treliça para este código convolucional é mostrada na Figura 4.13. Como a treliça completa é formada pela repetição de uma mesma seção (Figura 4.13), por simplicidade, também chamaremos de treliça uma única seção da treliça.

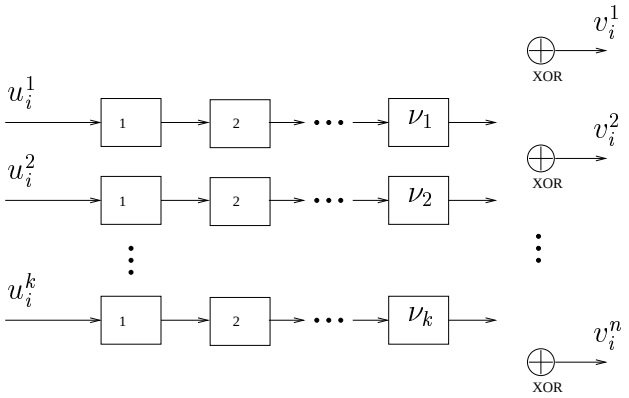


Figura 4.14: Codificador convolucional genérico de taxa $R = k/n$.

Os codificadores convolucionais mostrados até o momento contêm uma única entrada (um bit) de informaçao. De um modo mais geral, para um codificador convolucional de taxa $R = k/n$, temos o diagrama da Figura 4.14. As seqüências-código são obtidas pela mesma equação (4.14). Definimos o *comprimento de restrição da entrada* i, $1 \leq i \leq k$, denotado por ν_i, como o número de *flip-flops* associados à i-ésima entrada do codificador. O *comprimento de restrição total* do codificador é dado por:

$$\nu = \sum_{i=1}^{k} \nu_i$$

O número de estados na treliça será 2^{ν}. Note que a memória do código é dada por

$$m = \max_i \{\nu_i\}$$

Vejamos um exemplo.

Exemplo 61 *Vamos considerar um código convolucional de taxa $R = 2/3$ e de memória $m = 1$, descrito pelas seguintes matrizes geradoras:*

$$\mathbf{G}_0 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \quad \mathbf{G}_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

cujo codificador é mostrado na Figura 4.15.

A treliça para este código convolucional é mostrada na Figura 4.16.

4.2.1 O Algoritmo de Viterbi

A decodificação de máxima verossimilhança, ou seja, aquela que minimiza a probabilidade de erro de seqüência, para códigos convolucionais, é obtida utilizando-se o

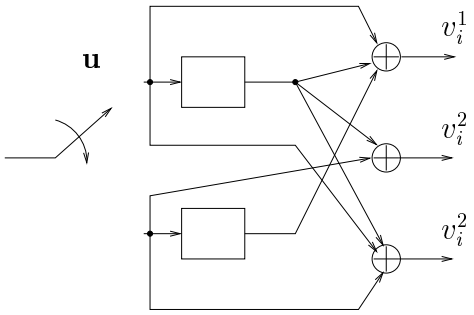


Figura 4.15: Codificador convolucional de taxa R=2/3, memória $m = 1$, $\nu_1 = 1$, $\nu_2 = 1$ e $\nu = 2$ do Exemplo 61.

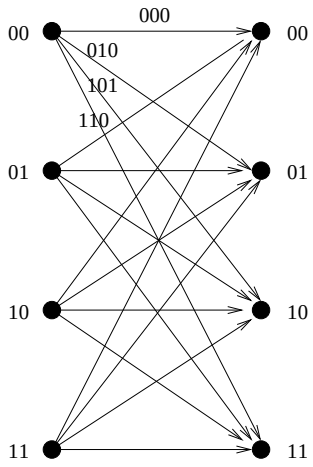


Figura 4.16: Treliça para o código convolucional de taxa R=2/3 do Exemplo 61.

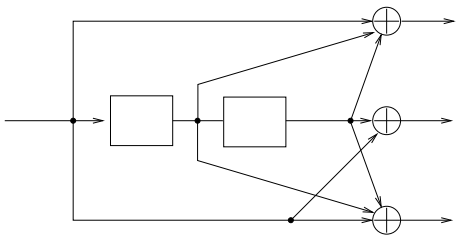


Figura 4.17: Codificador convolucional de taxa R=1/3, memória $m = 2$ e $\nu = 2$ do Exemplo 62.

algoritmo de Viterbi. O algoritmo de Viterbi pode ser descrito da seguinte maneira. Vamos supor que a seqüência recebida seja $\mathbf{r}$. Considerando a treliça para o código convolucional, o algoritmo calcula a distância de Hamming entre os primeiros n bits de $\mathbf{r}$ e o rótulo sobre cada ramo na primeira seção da treliça. Os valores destas distâncias são armazenados na forma de um rótulo sobre o próximo estado, aquele ao qual o ramo está conectado. O procedimento é repetido com relação aos próximos n bits de $\mathbf{r}$ e o rótulo sobre cada ramo na segunda seção da treliça, onde neste segundo passo, armazena-se no próximo estado o valor da distância acumulada, isto é, a distância armazenada no estado anterior somada à distância obtida com o ramo atual. Caso haja mais de um ramo chegando ao mesmo estado, aquele ramo que acumular neste próximo estado uma distância menor será o sobrevivente, descartando-se os demais. Se houver empate das distâncias acumuladas de dois ou mais caminhos chegando ao mesmo estado, então arbitrariamente, sem qualquer perda de desempenho, seleciona-se qualquer um dos caminhos com distância acumulada mínima como o sobrevivente, descartando-se os demais. Este procedimento é repetido até o final da treliça, onde restará um único estado, com um único caminho sobrevivente. A seqüência-código correspondente a este caminho será a seqüência decodificada (dada como correta). Vamos ilustrar a decodificação de códigos convolucionais por meio de um exemplo.

Exemplo 62 *Vamos considerar um código convolucional de taxa $R = 1/3$ e de memória $m = 2$, descrito pelas seguintes matrizes geradoras:*

$$\mathbf{G}_0 = [1 \;\; 1 \;\; 1] \;\; \mathbf{G}_1 = [1 \;\; 0 \;\; 1] \;\; \mathbf{G}_2 = [1 \;\; 1 \;\; 1]$$

cujo codificador é mostrado na Figura 4.17.

A treliça para este código convolucional é mostrada na Figura 4.18. Devemos notar que

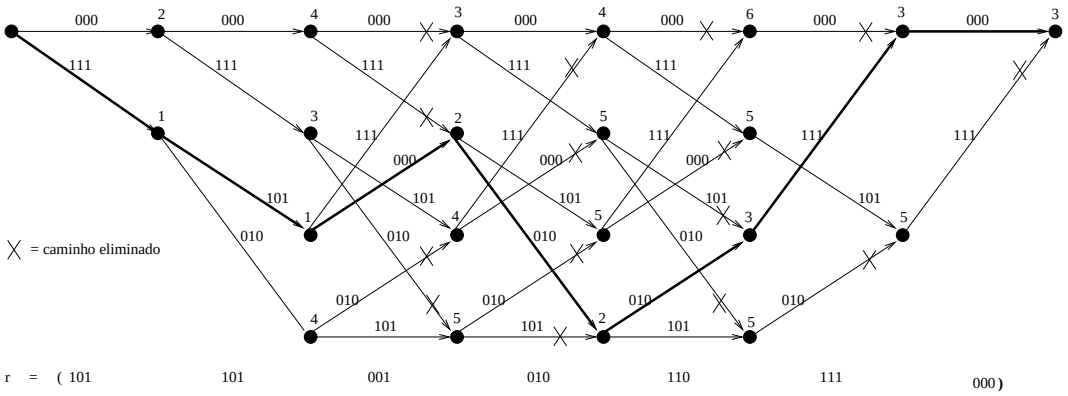


Figura 4.18: Ilustração do algoritmo de Viterbi com a treliça para o código convolucional de taxa R=1/3 do Exemplo 62.

para que no final da treliça tenhamos um único estado, é necessário entrarmos com mk zeros no codificador (m zeros para cada uma das k entradas), fazendo com que este volte ao estado zero. Vamos supor que a seqüência de informação seja $\mathbf{u} = [1, 0, 1, 1, 0]$. Para que o codificador retorne ao estado zero, devemos entrar com 2 zeros, que não fazem parte dos bits de informação. Assim, o vetor $\mathbf{u}$ modificado será: $\mathbf{u} = [1, 0, 1, 1, 0, 0, 0]$. A palavra-código correpondente é dada por:

$$\mathbf{v} = [111, 101, 000, 010, 010, 111, 000]$$

Vamos supor que a seqüência recebida seja

$$\mathbf{r} = [101, 101, 001, 010, 110, 111, 000]$$

ou seja, 3 erros de bit aconteceram na transmissão da seqüência $\mathbf{v}$. O algoritimo de Viterbi é ilustrado na Figura 4.18. Note que o algoritmo produz a decodificação correta de $\mathbf{r}$, que é o caminho indicado em negrito na treliça.

Capítulo 5

Probabilidade II: Variáveis Aleatórias Contínuas

5.1 Densidade de Probabilidades

No Capítulo 2, estudamos as variáveis aleatórias discretas, onde consideramos espaços amostrais finitos ou infinitos, porém numeráveis. Um exemplo de tal situação é obtido a partir do lançamento de um dado ou de uma moeda. Os possíveis resultados destes experimentos formam um conjunto finito e numerável.

Muitos dos fenômenos aleatórios, porém, apresentam resultados que são conjuntos de números reais. Por exemplo, a tensão $v(t)$, no instante t, medida nos terminais de um resistor ruidoso apresenta valores reais. O nível de brilho de um ponto particular na tela de um monitor, e a amplitude do ruído num dado instante em um sistema de comunicação também são números reais. Mesmo quando o espaço amostral Ω não for numérico, podemos gerar um novo espaço amostral a partir de Ω que seja numérico. Por exemplo, podemos atribuir números reais aos tons aleatórios de cinza de um pixel em uma imagem.

Neste capítulo, vamos considerar as situações descritas no último parágrafo, ou seja, vamos considerar as *variáveis aleatórias reais*. Para uma v.a. real X, teremos que o espaço amostral é dado por $\mathcal{R}$, o cojunto dos números reais.

5.1.1 Função Distribuiçao de Probabilidades (FDP)

A função distribuição de probabilidades (FDP), definida a seguir, se aplica tanto para variáveis aleatórias discretas quanto para variáveis aleatórias contínuas. Seja X uma variável aleatória. A FDP é definida como:

$$F_X(x) = P[X \leq x] = P_x[(-\infty, x]]$$

Se $F_X(x)$ for descontínua em um ponto, digamos x_0, então $F_X(x_0)$ será dada pelo valor da PDF imediatamente à direita de x_0, enquanto que $F_X(x_0^-)$ denotará o valor imediatamente à esquerda de x_0.

Propriedades de $F_X(x)$

1. $0 \leq F_X(x) \leq 1$
2. $F_X(-\infty) = 0$ e $F_X(\infty) = 1$
3. Se $x_1 \leq x_2$, então $F_X(x_1) \leq F_X(x_2)$, ou seja, $F_X(x)$ é uma função não decrescente de x.
4. Se $F_X(x)$ apresentar descontinuidade, então X é uma v.a. discreta. Se $F_X(x)$ for contínua, então X é uma v.a. real.

Exemplo 63 *Seja X uma v.a. discreta representando os resultados do lançamento de um dado. Assim, a FDP $F_X(x)$ é aquela mostrada na Figura 5.1.*

Exemplo 64 *Seja X uma v.a. que denota o instante de chegada de um ônibus à estação no intervalo $(0, T]$. Claramente, $F_X(t) = 0$ para $t \leq 0$ e $F_X(T) = 1$. Supondo-se que o instante de chegada do ônibus seja uniformemente provável no intervalo $(0, T]$, então a FDP de X é aquela mostrada na Figura 5.2.*

5.1.2 Função Densidade de Probabilidades (fdp)

A função densidade de probabilidades (fdp) é definda para variáveis aleatórias reais. Neste caso, $F_X(x)$ é contínua e a fdp é definida como:

$$f(x) = \frac{dF(x)}{dx},$$

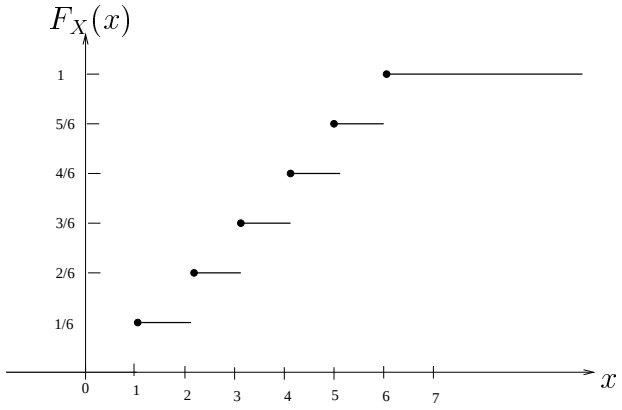


Figura 5.1: FDP da v.a. discreta representando os resultados do lançamento de um dado.

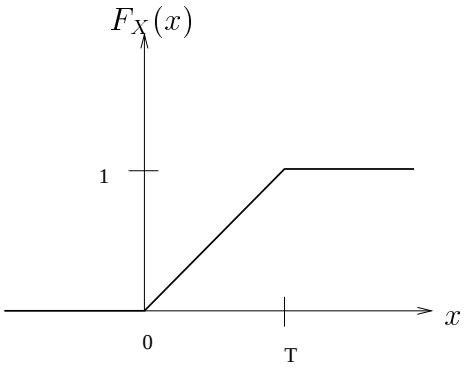


Figura 5.2: FDP da v.a. discreta representando o instante de chegada uniformemente distribuído no intervalo $(0, T]$.

onde $F(x) \triangleq F_X(x)$.

Propriedades de $f(x)$

1. $f(x) \geq 0$
2. $\int_{-\infty}^{\infty} f(\xi)\, d\xi = F_X(\infty) - F_X(-\infty) = 1$
3. $F(x) = \int_{-\infty}^{x} f(\xi)\, d\xi = P[X \leq x]$

Note que

$$\begin{aligned} P[x_1 < X \leq x_2] &= P[X \leq x_2] - P[X \leq x_1] \\ &= F_X(x_2) - F_x(x_1) \\ &= \int_{x_1}^{x_2} f(x)\, dx, \end{aligned}$$

daí o nome densidade de probabilidades.

Como o número de possibilidades para o valor de uma v.a. contínua é infinito, é plausível acreditarmos que a probabilidade de uma variável aleatória contínua X assumir um certo valor específico seja zero! (Tente encontrar uma agulha num palheiro.) De fato,

$$P[X = x_0] = P[x_0 < X \leq x_0] = P[X \leq x_0] - P[X \leq x_0] = F_X(x_0) - F_X(x_0) = 0$$

5.1.3 Exemplos de Densidade de Probabilidades

Algumas exemplos de fdp's de variáveis aleatórias contínuas são:

1. Uniforme:

$$f(x) = \begin{cases} \frac{1}{b-a}, & a \leq x \leq b \\ 0, & \text{fora} \end{cases}$$

onde a e $b > a$ são números reais. Esta densidade é usada quando não se tem qualquer informação *a priori* a respeito da distribuição de probabilidades dos resultados. Por exemplo, a fase de um sinal recebido em um sistema de comunicações pode ser considerada uniformemente distribuída no intervalo $[0, 2\pi)$.

2. Gaussiana:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$$

onde μ e $\sigma^2 \geq 0$ são números reais que representam a média e a variância, respectivamente. Essa densidade ocorre em vários ramos da ciência e da engenharia. Por exemplo, ela serve de modelo para um ruído aditivo em um sistema de comunicações: o ruído Gaussiano.

3. Exponencial ($\mu > 0$):

$$f(x) = \frac{1}{\mu} e^{-x/\mu} u(x),$$

onde $u(x)$ é a função degrau. Esta densidade ocorre, por exemplo, em problemas de tempo de espera em filas e de tempo de vida útil de uma máquina.

4. Rayleigh ($\sigma > 0$):

$$f(x) = \frac{x}{\sigma^2} e^{-x^2/2\sigma^2} u(x).$$

Esta densidade modela, por exemplo, as flutuações da envoltória de um sinal transmitido por um canal de rádio-móvel, o chamado canal com desvanecimento (*Rayleigh fading*).

5.1.4 Momentos de uma Variável Aleatória Contínua

No Capítulo 2, vimos que a média e a variância trazem alguma informação a respeito da variável aleatória. Naturalmente, estes dois parâmetros estão bem longe de descrever completamente o comportamento da variável aleatória. Por exemplo, duas variáveis aleatórias podem ter a mesma média, mas variâncias diferentes. Ou a mesma média e a mesma variância, mas distribuições ou densidades de probabilidades distintas. Parâmetros como média, variância e valor quadrático médio são exemplos de *momentos*. Um variável aleatória pode ter vários momentos de ordens mais elevadas e, sob certas condições, é possível descrever completamente o comportamento da v.a., isto é, reconstruir a sua fdp a partir de *todos* os seus momentos.

O r-ésimo *momento* de uma variável aleatória contínua X é definido como:

$$\xi_r \triangleq E\{X^r\} = \int_{-\infty}^{\infty} x^r f(x)\, dx, \quad \text{onde } r = 0, 1, 2, 3, \ldots$$

Para $r = 1$, temos que ξ_1 é a média, e para $r = 2$, temos que ξ_2 é o valor quadrático médio. A varância é dada por $\xi_2 - (\xi_1)^2$.

O r-ésimo *momento central* de uma variável aleatória contínua X é definido como:

$$m_r \triangleq E\{(X-\mu)^r\} = \int_{-\infty}^{\infty} (x-\mu)^r f(x)\, dx, \quad \text{onde } r = 0, 1, 2, 3, \ldots$$

Para $r = 1$, temos que $m_1 = 0$. m_2 é a varância. Por exemplo, para a densidade uniforme,

$$\xi_1 = E\{X\} = \int_{-\infty}^{\infty} x f(x)\, dx = \int_a^b \frac{x}{b-a} = \frac{a+b}{2}$$

Para a densidade Gaussiana,

$$\xi_1 = E\{X\} = \mu, \qquad m_2 = E\{(X-\mu)^2\} = \sigma^2.$$

5.1.5 A Lei dos Grandes Números

A Lei dos Grandes Números trata da convergência de uma seqüência de estimativas da média de uma variável aleatória. Seja $X[n]$ uma seqüência de variáveis aleatórias estatisticamente independentes e identicamente distribuídas, com média μ_x e variância σ_X^2, definida para $n \geq 1$. Defina uma outra seqüência aleatória como:

$$\hat{\mu}_X[n] \triangleq \frac{1}{n} \sum_{k=1}^{n} X[k] \qquad \text{para } n \geq 1.$$

Então, quando $n \to \infty$, temos que $\hat{\mu}_x[n] \to \mu_X$ em probabilidade, ou seja, para todo $\epsilon > 0$ temos que

$$\lim_{n\to\infty} P[|\hat{\mu}_x[n] - \mu_X| > \epsilon] = 0.$$

Na prática, a Lei dos Grandes Números pode ser usada para se estimar a média de uma variável aleatória a partir de um número grande n de experimentos. Por exemplo, para se estimar a intensidade média de um sinal de rádio em uma dada posição, pode-se realizar n medições através de equipamentos especializados e, se o número de medições for suficientemente grande, a Lei dos Grandes Números garante que a probabilidade de a média amostral obtida acima ser muito próxima da média verdadeira é próxima de 1.

5.1.6 A Variável Aleatória Normal

Vamos considerar a densidade Gaussiana com média μ e variância σ^2 apresentada na Seção 5.1.3, repetida aqui por conveniência:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$$

Estamos interessados em calcular a probabilidade de uma variável aleatória Gaussiana X com densidade $f(x)$ assumir um valor maior ou igual a um certo valor específico. Ou seja, desejamos obter $P[X \geq a]$, onde a é um número real qualquer. Pela terceira propriedade da fdp, vista na Seção 5.1.2, teremos que

$$\begin{aligned} P[X \geq a] &= \int_a^\infty f(\xi)\, d\xi \\ &= \frac{1}{\sigma\sqrt{2\pi}} \int_a^\infty \exp\left[-\frac{1}{2}\left(\frac{\xi-\mu}{\sigma}\right)^2\right]\, d\xi. \end{aligned}$$

Devemos agora fazer a seguinte mudança de variável:

$$z = \frac{\xi - \mu}{\sigma}.$$

Assim, teremos a seguinte integral:

$$\begin{aligned} P[X \geq a] &= \frac{1}{\sqrt{2\pi}} \int_{\frac{a-\mu}{\sigma}}^\infty \exp\left[-\frac{z^2}{2}\right]\, dz \qquad (5.1) \\ &\triangleq Q\left(\frac{a-\mu}{\sigma}\right). \end{aligned}$$

O integrando da integral (5.1) é uma densidade Gaussiana com média zero e variância 1. Esta densidade é conhecida como a *densidade normal*. A função $Q(\alpha)$ é conhecida como a *função de erro complementar* ou *função Q*, e seus valores são obtidos a partir de tabelas. A equação (5.1) é muito utilizada nos cálculos de probabilidade de erro de modulações digitais quando o canal de comunicação introduz ruído Gaussiano (ver Capítulo 6).

5.1.7 O Teorema do Limite Central

O Teorema do Limite Central (TLC) estabelece que a suma de um grande número de variáveis aleatórias (independentemente de suas densidades individuais) tende para

uma variável aleatória normal. Este resultado é de suma importância e tem utilização em diversas áreas da ciência e da engenharia. Por exemplo, como veremos no Capítulo 6, o TLC justifica a adoção do modelo de densidade Gaussiana para o ruído térmico, visto que este é resultado da soma dos efeitos da agitação de um número infinitamente grande de elétrons. O TLC, numa forma mais simples, pode ser apresentado da seguinte maneira.

Teorema 11 *Sejam $X_1, X_2, \ldots, X_n$ variáveis aleatórias mutuamente independentes com FDP's $F_1(x_1), F_2(x_2), \ldots, F_n(x_n)$, respectivamente, tais que, para $1 \leq k \leq n$,*

$$\mu_k = 0, \qquad \sigma_k^2 = 1$$

Então, a soma normalizada

$$Z_n \stackrel{\Delta}{=} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} X_i$$

converge para a densidade normal, quando $n \to \infty$.

Uma forma mais geral do TLC admite que as variâncias sejam distintas ou dependentes de k, mas certas condições matemáticas devem neste caso ser satisfeitas para garantir a convergência.

Capítulo 6

Processos Estocásticos

Um *processo estocástico* ou *processo aleatório* é uma extensão do já apresentado conceito de variável aleatória. Quando consideramos variáveis aleatórias, associamos um número real a cada possível resultado do experimento aleatório. Em um processo estocástico, estaremos associando uma função do tempo a cada possível resultado do experimento aleatório. Por exemplo, no caso do lançamento de um dado, associamos os números 1, 2, 3, 4, 5 e 6 aos possíveis resultados do lançamento. Seja $\xi_i \in \{1, 2, 3, 4, 5, 6\}$ o resultado de um lançamento do dado. Podemos agora associar a este resultado a função $\xi_i e^{-t} u(t)$, por exemplo. Assim, para um instante de tempo específico, digamos $t = 1$, teremos uma variável aleatória; neste caso os possíveis valores da variável aleatória são $e^{-1}, 2e^{-1}, \ldots, 6e^{-1}$, cada um ocorrendo com probabilidade $\frac{1}{6}$. Para um ξ_i específico, digamos $\xi_i = 3$, teremos simplesmente uma função determinística no tempo: $3e^{-t}u(t)$. Se fixarmos t e ξ concomitantemente, teremos um número real (por exemplo, para $t = 1$ e $\xi = 3$ teremos o número real $3e^{-1} = 1,1036\ldots$). Este exemplo considera uma variável aleatória discreta. Podemos definir um processo estocástico baseado numa variável contínua.

De um modo mais geral, um processo estocástico pode ser visto como uma coleção de funções (ou sinais) do tempo correpondendo a vários resultados de um experimento aleatório. Assim, associado a cada resultado ξ_i do experimento aleatório em questão, teremos uma função determinística do tempo: $X(t, \xi_i)$, chamada de *função amostra* do processo estocástico. Num instante de tempo específico, digamos t', ou seja

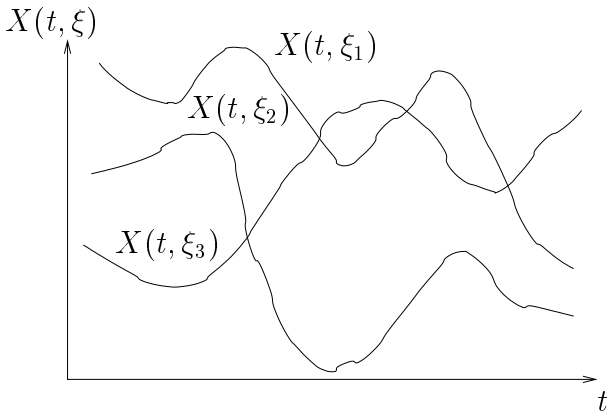


Figura 6.1: Ilustração de um processo estocástico.

amostrando o processo estocástico $X(t,\xi)$ no instante t', teremos a variável aleatória $X(t',\xi)$, enquanto que $X(t',\xi_i)$ correponderá a um número real. Uma ilustração de um processo estocástico é fornecida na Figura 6.1. Para siimplificar a notação, tornaremos implícita a dependência do processo estocástico com relação a ξ. Assim, o processo estocástico será denotado simplesmente por $X(t)$.

Teremos uma descrição completa de um processo estocástico $X(t)$ se para qualquer inteiro n e para qualquer escolha de $(t_1, t_2, \ldots, t_n) \in \mathcal{R}^n$, a fdp conjunta de $(X(t_1), X(t_2), \ldots, X(t_n))$, ou seja, $f_{X(t_1),X(t_2),\ldots,X(t_n)}(x_1, x_2, \ldots, x_n)$ for conhecida.

6.1 Médias Estatísticas de um Processo Estocástico

Seja $X(t)$ um processo estocástico. Num dado instante de tempo $t = t_k$, sabemos que $X(t_k)$ representa uma variável aleatória obtida a partir da amostragem de $X(t)$ no instante t_k. Definimos a *média* de um processo estocástico como:

$$E\{X(t_k)\} = \int_{-\infty}^{\infty} x\, f_{X_k}(x)\, dx = m_X(t_k)$$

Note que para um t genérico, temos que a média $m_X(t)$ do processo estcástico é uma função do tempo.

Considere agora duas variáveis aleatórias $X(t_1)$ e $X(t_2)$ obtidas a partir da amostragem de $X(t)$ nos instantes t_1 e t_2, respectivamente. Assim, definimos a *autocorrelação* do processo estocástico como:

$$R_X(t_1, t_2) = E\{X(t_1)\, X(t_2)\}$$

A correlação mede o quanto as variáveis aleatórias $X(t_1)$ e $X(t_2)$ são correlacionadas.

6.2 Estacionaridade

Em muitas situações práticas, não é possível ou não é necessário se ter a descrição completa de um processo estocástico, ou seja, o conhecimento de todas as fdp's conjuntas citadas acima. Em muitos destes casos práticos, podemos supor certas condições de estacionaridade, possibilitando a obtenção de informações importantes a respeito do processo.

Um processo $X(t)$ é dito ser *estacionário no sentido estrito* se nenhuma das estatísticas do processo é afetada por um deslocamento no tempo. Mais especificamente, para todo inteiro n, para quaisquer instantes de tempo $(t_1, t_2, \ldots, t_n) \in \mathcal{R}^n$, e para qualquer $\Delta > 0$,temos que

$$f_{X(t_1),X(t_2),\ldots,X(t_n)}(x_1, x_2, \ldots, x_n) = f_{X(t_1+\Delta),X(t_2+\Delta),\ldots,X(t_n+\Delta)}(x_1, x_2, \ldots, x_n)$$

A estacionaridade no sentido estrito é uma condição muito forte. Em muitas situações práticas, faz-se necessária a definição de uma forma menos restritiva de estacionaridade. Um processo $X(t)$ é dito ser *estacionário no sentido amplo* se as seguintes condições são satisfeitas:

1. $m_X(t) = m_X$, ou seja, a média não depende de t.
2. $R_X(t_1, t_2) = R_X(t_2 - t_1) = R_X(\tau)$, onde $\tau = t_2 - t_1$. Ou seja, a autocorrelação depende apenas da diferença entre os dois instantes de amostragem, e não dos valores de t_1 e de t_2 individualmente.

Em comunicações, normalmente supomos que os processos envolvidos são estacionário no sentido amplo. O exemplo mais importante é o processo que modela o ruído Gaussiano.

Um processo é dito ser um *processo Gaussiano* se todas as fdp's (individuais e conjuntas) do processo são Gaussianas. Temos o seguinte resultado, cuja prova é omitida.

Teorema 12 *Todo processo estacionário no sentido estrito é estacionário no sentido amplo. A recírpoca nem sempre é verdadeira, a menos que o processo seja Gaussiano. Ou seja, nem todo processo estacionário no sentido amplo é estacionário no sentido estrito; um processo Gaussiano estacionário no sentido amplo é estacionário no sentido estrito.*

6.3 Propriedades da Autocorrelação de um Processo Estocástico

A autocorrelação de um processo $X(t)$ estacionário no sentido amplo tem as seguintes propriedades:

1. $R_X(\tau) = R_X(-\tau)$
2. $R_X(\tau) \leq R_X(0)$, para todo τ.
3. A transformada de Fourier de $R_X(\tau)$ é a *densidade espectral de potência* $G_X(f)$ do processo, ou seja:

$$G_X(f) = \int_{-\infty}^{\infty} R_X(\tau) e^{-j2\pi f\tau}\, d\tau$$

4. $R_X(0) = E\{X^2(t)\}$ (valor quadrático médio).

6.4 Propriedades da Densidade Espectral de Potência de um Processo Estocástico

A densidade espectral de potência de um processo $X(t)$ estacionário no sentido amplo tem as seguintes propriedades:

1. $G_X(f) \geq 0$
2. $G_X(f) = G_X(-f)$, para $X(t)$ real

3. A transformada inversa de Fourier de $G_X(f)$ é $R_X(\tau)$, ou seja:

$$R_X(\tau) = \int_{-\infty}^{\infty} G_X(f) e^{j2\pi f\tau}\, df$$

4. A potência média total de $X(t)$ é dada por:

$$P_X = \int_{-\infty}^{\infty} G_X(f)\, df$$

6.5 Ruído em Sistemas de Comunicações

Em sistemas de comunicações, a forma mais relevante de ruído é o chamado *ruído térmico* ou *ruído Johnson*. Este ruído é modelado por um processo estocástico Gaussiano, estacionário no sentido amplo, de média zero. Ou seja, para qualquer instante de tempo t' específico, a variável aleatória $n(t')$ (ou simplesmente n) tem fdp igual a:

$$f(n) = \frac{1}{\sigma\sqrt{2\pi}} exp\left[-\frac{1}{2}\left(\frac{n}{\sigma}\right)^2\right],$$

onde σ^2 é a variância de n. O ruído Gaussiano é adicionado ao sinal transmitido. Seja a um nível de tensão correspondente ao símbolo transmitido a partir de uma modulação digital em banda básica. Assim, o sinal (real) recebido z será uma variável aleatória dada por:

$$z = a + n$$

A v.a. z é Gaussiana com média a e variância σ^2, ou seja:

$$f(z) = \frac{1}{\sigma\sqrt{2\pi}} exp\left[-\frac{1}{2}\left(\frac{z-a}{\sigma}\right)^2\right],$$

As características espectrais do processo estocástico que modela o ruído são dadas por uma densidade espectral de potência constante, cujo valor é denotado por $\frac{N_o}{2}$, para todas as freqüências, ou seja:

$$G_n(f) = \frac{N_o}{2} \qquad \text{(Watts/Hz)}$$

Esta densidade é mostrada na Figura 6.2(a). Em analogia à luz branca, que contém todos os comprimentos de onda do espectro da luz vizível na mesma intensidade, este ruído em comunicações é denominado de *ruído branco*. Muito freqüentemente, na

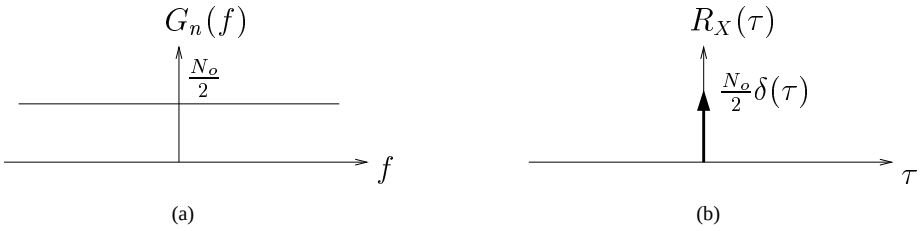


Figura 6.2: (a) Densidade espectral de potência e (b) função de autocorrelação do ruído branco.

literatura, este ruído é referido como AWGN, do inglês "Additive White Gaussian Noise".

Devemos notar que a potência do ruído branco é infinita! Pela propriedade 4) da densidade espectral de potência, a potência média total do ruído branco é dada por:

$$P_n = \int_{-\infty}^{\infty} G_X(f)\, df = \int_{-\infty}^{\infty} \frac{N_o}{2}\, df = \infty$$

Contornamos este problema em comunicações através de uma filtragem, como veremos mais adiante.

A função de autocorrelação do ruído branco é, pela propriedade 3) da densidade espectral de potência, dada por:

$$R_X(\tau) = \int_{-\infty}^{\infty} G_X(f) e^{j2\pi f\tau}\, df = \int_{-\infty}^{\infty} \frac{N_o}{2} e^{j2\pi f\tau}\, df = \frac{N_o}{2}\delta(\tau)$$

que é mostrada na Figura 6.2(b). Note, a partir de $R_X(\tau)$, que amostras distintas (obtidas em instantes de tempo distintos) do ruído são descorrelacionadas, pois $R_X(\tau) = 0$ para $\tau > 0$. Como o ruído é Gaussiano, temos que amostras distintas são estatisticamente independentes. Em outras palavras, o ruído afeta cada símbolo transmitido independentemente, o que significa que o canal não tem *memória*.

6.6 Transmissão de Sinais através de Sistemas Lineares

Nas seções anteriores, estudamos os modelos para sinais e ruído. Nesta seção, estudaremos a caracterização de sistemas (ou filtros) lineares, e o efeito destes sobre os sinais. Se um sinal $x(t)$ contínuo no tempo for processado por um filtro linear, como será o sinal

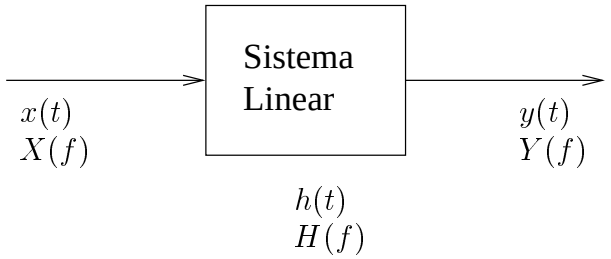


Figura 6.3: Sistema linear e sinais de entrada e de saída.

de saída deste filtro? Analisaremos este efeito tanto no domínio do tempo quanto no domínio da freqüência, usando propriedades da transformada de Fourier. No domínio do tempo, o sinal de saída, $y(t)$, será determinado a partir do sinal de entrada, $x(t)$, e da função *resposta ao impulso* do sistema, $h(t)$. No domínio da freqüência, teremos o sinal de saída, $Y(f)$, determinado a partir do sinal de entrada, $X(f)$, e da *função de transferência* do sistema, $H(f)$, onde $X(f)$, $Y(f)$ e $H(f)$ são, respectivamente, as transformadas de Fourier de $x(t)$, $y(t)$ e $h(t)$. Deixamos para apresentar o efeito de um filtro linear sobre um processo estocástico na próxima seção.

Um sistema linear com resposta ao impulso $h(t)$ (função de transferência $H(f)$), sinal de entrada $x(t)$ ($X(f)$), e sinal de saída $y(t)$ ($Y(f)$), é mostrado na Fingura 6.3.

6.6.1 A Resposta ao Impulso

Um sistema linear e invariante no tempo é caracterizado pela resposta ao impulso $h(t)$, que é a resposta (sinal de saída) quando a entrada do sistema for a função impulso de Dirac $\delta(t)$, ou seja,

$$h(t) = y(t) \quad \text{quando } x(t) = \delta(t)$$

Se a entrada for um sinal genérico $x(t)$, a saída é obtida pela convolução:

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)\, d\tau \tag{6.1}$$

Um sistema é dito ser *causal* se $h(t) = 0$ para $t < 0$. Sistemas reais ou fisicamente realizáveis são causais.

6.6.2 A Função de Transferência

Podemos aplicar a transformada de Fourier na equação (6.1), obtendo assim a relação entre entrada e saída no domínio da freqüência:

$$Y(f) = X(f)\, H(f) \qquad \Rightarrow \quad H(f) = \frac{Y(f)}{X(f)} \tag{6.2}$$

que resulta na função de transferência $H(f)$.

Podemos escrever $H(f)$ na forma:

$$H(f) = |H(f)|e^{j\theta(f)}$$

onde

$$|H(f)| = H(f)\, H^*(f)$$

é a *resposta em amplitude* do sistema e

$$\theta(f) = \tan^{-1}\left[\frac{\mathrm{Im}\{H(f)\}}{\mathrm{Re}\{H(f)\}}\right]$$

é a *resposta da fase*. Se $h(t)$ for uma função real, então

$$|H(f)| \text{ é uma função par } \Rightarrow |H(f)| = |H(-f)|$$

e

$$\theta(f) \text{ é uma função ímpar } \Rightarrow \theta(f) = -\theta(-f)$$

6.6.3 Transmissão Sem Distorção: Filtro Ideal

Em algumas situações práticas, podemos caracterizar um canal de comunicação por um sistema linear invariante no tempo, com resposta ao impuso $h(t)$ (função de tranferência $H(f)$). Estamos interessado em obter as características de um sistema que é capaz de processar o sinal sem provocar nele nenhuma distorção. Mais especificamente, permitiremos que o sinal de entrada apenas sofra um atraso t_0 e que seja atenuado (ou amplificado) por um fator K, mas que as suas características principais sejam preservadas pelo sistema. Em outras palavras, se o sinal de entrada for $x(t)$, então gostaríamos que o sinal de saída fosse dado por:

$$y(t) = K\, x(t - t_0)$$

onde t_0 e K são constantes reais positivas.

Podemos aplicar a transformada de Fourier na equação acima, e obtemos:

$$Y(f) = K\ X(f)e^{-j2\pi f t_0}$$

de onde tiramos que a função de transferência deve ser da forma:

$$H(f) = \frac{Y(f)}{X(f)} = K\ e^{-j2\pi f t_0} \tag{6.3}$$

Ou seja, o sistema ideal é aquele que tem resposta em em aplitude plana (constante):

$$|H(f)| = K, \quad \text{para toda freqüência } f$$

e fase linear:

$$\theta(f) = -2\pi f t_0 \quad \text{(radianos)}$$

Se para um canal de comunicações $H(f)$ tivermos que $|H(f)| \neq K$ e/ou $\theta(f)$ não variar linearmente com a freqüência, então haverá necessaidade de *equalização*.

6.7 Processos Estocásticos e Sistemas Lineares

Seja $X(t)$ um processo estocástico estacionário no sentido amplo. Considere um sistema linear invariante no tempo com resposta ao impulso $h(t)$ (função de transferência $H(f)$). Se $G_X(f)$ for a densidade espectral de potência do processo $X(t)$, então a densidade espectral de potência do processo de saída, $Y(t)$, é dada por:

$$G_Y(f) = G_X(f)\ |H(f)|^2 \tag{6.4}$$

Podemos agora, a partir da equação (6.4), determinar a densidade espectral de potência do ruído branco $n(t)$ quando filtrado por um filtro passa-baixas ideal com função de transferência:

$$H(f) = \begin{cases} 1, & \text{para } |f| < W \\ 0, & \text{para } |f| \geq W \end{cases}$$

O ruído branco $n(t)$ tem densidade espectral de potência $G_n(f) = \frac{n_0}{2}$. Seja $\tilde{n}(t)$ o processo que representa o ruído na saída do filtro passa-baixas, e $G_{\tilde{n}}(f)$ a sua densidade

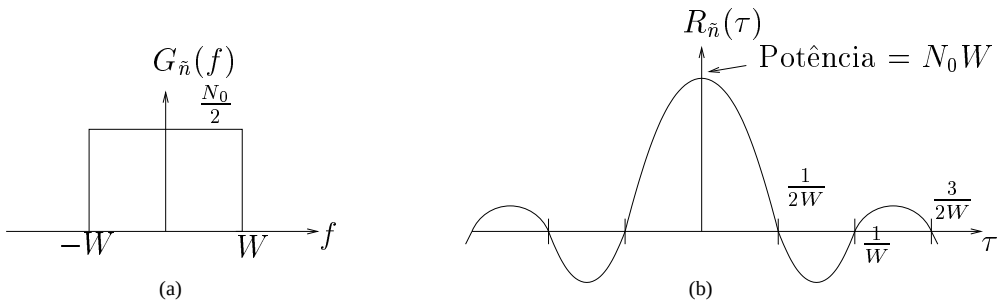


Figura 6.4: Ruído colorido.

espectral de potência. Assim, pela equação (6.4), temos que:

$$\begin{aligned} G_{\tilde{n}}(f) &= G_n(f)\,|H(f)|^2 \\ &= \frac{n_0}{2}\,|H(f)|^2 \\ &= \begin{cases} \frac{N_0}{2}, & \text{para } |f| < W \\ 0, & \text{para } |f| \geq W \end{cases} \end{aligned}$$

que é mostrada na Figura 6.4(a). Este é conhecido como o *ruído colorido*. A partir da propriedade 3) da densidade espectral de potência, temos que a função de autocorrelação do ruído colorido é:

$$R_{\tilde{n}}(\tau) = N_0\, W \frac{\sin(2\pi W\tau)}{2\pi W\tau}$$

que é mostrada na Figura 6.4(b). Note que duas amostras do ruído colorido separadas por um intervalo de tempo múltiplo de $\frac{1}{2W}$ são descorrelacionadas:

$$R_{\tilde{n}}(\tau) = 0, \qquad \text{para } \tau = \frac{k}{2W}, \quad k \in \mathcal{Z}$$

Se essas amostras forem variáveis aleatórias Gaussianas, então elas também serão independentes. Esta propriedade é importante para os receptores de sistemas de comunicações.

Por fim, devemos mencionar mais uma propriedade de processos estocásticos Gaussianos. Se um processo estocástico Gaussiano $X(t)$ for processado por um sistema linear, então o processo correspondendo à saida do sistema também será Gaussiano.