UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE TELEMÁTICA

CODIGOS DE TRELIÇA FIXOS E VARIANTES NO TEMPO

PROF. DR. REGINALDO PALAZZO JÚNIOR

Tese apresentada à FEE-UNICAMP como parte dos requisitos exigidos para obtenção do título de Prof. Livre-Docente na Área de Teoria de Informação e Codificação.

- 1987 -

"The longest journey starts with but
a single step"

Lao Tzu

À minha esposa Cristina e
minha filha Patrícia

AGRADECIMENTOS

Gostaria de agradecer à todos que, de uma forma ou de outra, me ajudaram, apoiaram e contribuíram decisivamente para que mais este degrau pudesse ser por mim galgado.

Em especial agradeço à Srta. Elza Aoki pela presteza, compreensão e dedicação exemplares sem as quais seguramente o que segue nestas páginas não teria finalidade.

RESUMO

Neste trabalho procuramos estabelecer os conceitos e métodos utilizados no desenvolvimento das pesquisas sobre códigos de treliça dando um novo enfoque aos problemas: a) de determinação dos códigos ótimos invariantes no tempo sobre o corpo de Galois com q elementos, sob o critério de mínima Pb através da equivalência com o problema de otimização combinatorial; b) de determinação dos códigos ótimos variantes no tempo, sob o critério minimização da função enumeradora; c) equivalência do problema tratado em a) com aplicações em criptografia; d) determinação da função enumeradora (generalizada) para códigos de treliça não lineares com aplicações em Múltiplo Acesso, Modulação por Sobreposição, Resposta Parcial, Modulação por Codificação de Treliça e etc....

ÍNDICE

CAPÍTULO 1

INTRODUÇÃO À CÓDIGOS DE ÁRVORE, DE TRELIÇA E CONVOLUCIONAIS

1.1 - INTRODUÇÃO

O que se observa na literatura é que sob o ponto de vista de aplicação, códigos convolucionais são amplamente utilizados com o objetivo de proporcionar *ganhos de codificação*. Por outro lado, é evidente a falta de uma estrutura matemática que possa ser utilizada de uma forma sistemática na análise de tais códigos. Assim sendo, esforços tem sido dispendidos por inúmeros pesquisadores no sentido de estabelecer uma estrutura algébrica conveniente, como ocorre com os códigos de bloco.

Recentemente, esforços na direção de formalizar rigorosamente tal estrutura tem sido frutíferos com o emprego do conceito de "Lattice" da Álgebra. Entretanto, este conceito, embora muito promissor, necessita de um período de amadurecimento para responder questões sobre propriedades estruturais de códigos convolucionais ainda em aberto.

Deste modo, o objetivo deste capítulo é o de introduzir conceitos e definições de classes de códigos tais como, códigos de árvore, códigos de treliça e códigos convolucionais. Tais conceitos são necessários no sentido de padronização e uniformização dos fundamentos dessas classes de códigos tão importantes.

1.2 - CÓDIGOS DE ÁRVORE

Dentre as classes de grafos, a classe das árvores é bastante importante para a conceituação dos códigos que serão descritos a seguir. Uma árvore $A = (D, M)$ é um grafo conectado sem ciclos onde, para nossa conveniência, D especifica a profundidade com μ ramos divergindo de cada vértice e M é a profundi

dade onde somente um ramo sai de cada vértice.

Mais formalmente, definiremos uma árvore (D, M) μ-ária como sendo uma árvore tal que: 1) μ ramos divergem de cada vértice até uma profundidade D do vértice inicial; 2) e que somente um ramo diverge de cada vértice com profundidade maior ou igual a D, porém, menor ou igual a D+M, com D, M e μ inteiros tal que $D \geq 1$, $M \geq 0$ e $\mu \geq 2$.

Note que em geral uma árvore (D, M) μ-ária possuem μ^D vértices terminais com profundidade D+M.

À cada ramo desta árvore associaremos uma "palavra de ramo" com N símbolos do alfabeto de entrada de um canal discreto sem memória.

Seja R a taxa do código de árvore (D, M) μ-ária, então $\mu = 2^{NR}$ é o número de ramos que divergem de cada vértice até uma profundidade D.

Seja γ o comprimento de restrição da árvore, isto é, o número total de dígitos desde o vértice D-1 até D+M. Dessa forma, γ é dado por

$$\gamma = (M + 1) \cdot N$$

Defina como um código de árvore a quartupla (N, R, D, M). Note que este código forma uma classe especial dos códigos de bloco. O código de árvore (N, R, D, M) considerado como um código de bloco consiste de μ^D palavras cada tendo comprimento (D+M) x N. A taxa deste código é dada por

$$\tilde{R} = \frac{\log(\mu^D)}{(D+M) \cdot N} = \frac{D \log \mu}{(D+M) \cdot N}$$

como $\mu = 2^{NR}$, então

$$\tilde{R} = \left(\frac{D}{D+M}\right) \cdot R$$

como em geral $D >> M$ então $\tilde{R} \cong R$.

1.3 - CÓDIGOS DE TRELIÇA

Esta classe de códigos muito importante pertence à classe dos códigos de árvores pela introdução da dependência entre os símbolos da fonte no processo de codificação.

A introdução da dependência gera a necessidade de se estabelecer um procedimento de codificação para o código de árvore. Nessa direção, seja $\{\alpha_i\}_{i=o}^{D-1}$ uma sequência de dígitos de informação μ-ária. Dessa forma, à cada dígito estará sendo associado um ramo que diverge de cada vértice até a profundidade D. À cada sequência $\alpha_o, \alpha_1, \alpha_2, \ldots, \alpha_{D-1}$ estará sendo associado um caminho nesta árvore até a profundidade D.

Suponha que para algum inteiro ν entre M e D + M inclusive, rotulemos cada vértice da árvore (D, M) μ-ária com os ν dígitos de informação anteriores, também suponha que somente o dígito "0" será associado à cada vértice entre D e D + M. Dessa forma, fica estabelecido o esquema de codificação de um código de árvore (D, M, ν) μ-ária onde os vértices são enumerados.

Se esta árvore tem memória ν, no sentido de que para quaisquer dois vértices na mesma profundidade resultem na mesma sequência codificada restante quando a mesma sequência de dígitos de informação é aplicada à entrada do codificador a partir de quaisquer um desses vértices, então precisaremos manter so-

mente uma dessas sequências. Quando isto ocorre, temos o que é chamado de uma treliça (D, N, ν) μ-ária.

A quíntupla (N, R, D, M, ν) define um código de treliça para um canal discreto sem memória onde à cada ramo desta treliça associa-se uma "palavra de ramo" consistindo de N símbolos do alfabeto de entrada do canal.

Como (N, R, D, M, ν) é uma classe especial de (N, R, D, M) a restrição de memória γ continua sendo válida, o mesmo acontecendo com a taxa do código.

1.4 - CÓDIGOS CONVOLUCIONAIS

Pode-se mostrar via o famoso argumento "ensemble average" utilizado por Shannon na demonstração do Teorema Fundamental de Teoria de Informação que o "ensemble" dos códigos de árvores (N, R, D, M) para um canal discreto sem memória coincide com o "ensemble" dos códigos de treliça (N, R, D, M, ν = D + M). Assim sendo, é suficiente levarmos em consideração somente o "ensemble" dos códigos de treliça.

Seja $\{\alpha_i\}_{i=o}^{D-1}$ a sequência de dígitos de informação a ser codificada onde cada α_i está associado a um ramo da treliça a ser seguido pela sequência até a profundidade D. Seja $\{\beta_i\}_{i=o}^{D+M-1}$ a correspondente sequência codificada onde $\alpha_i = 0$ para $D \leq i \leq D+M-1$.

A representação matemática do codificador, segundo este modelo, é dada por

$$\beta_i = \Gamma(\alpha_i, \alpha_{i+1}, \ldots, \alpha_{i-\nu}, i)$$

Note que $\Gamma(\cdot)$ é uma função arbitrária dos ν valores anteriores a α_i e do tempo caracterizado por $t = i$. Quando β_i é dado por

$$\beta_i = \Gamma(\alpha_i, \alpha_{i-1}, \ldots, \alpha_{i-\nu})$$

o codificador é dito invariante no tempo. Observe que $\Gamma(\cdot)$ sendo arbitrária implica que a mesma pode ser uma função linear bem como não linear.

Os códigos convolucionais formam uma classe especial da classe dos códigos de treliça, quando $\Gamma(\cdot)$ é uma função linear.

De uma maneira mais formal, suponha que o alfabeto de entrada de um canal discreto sem memória é formado pelo corpo de Galois com q elementos, GF(q), onde q é um número primo ou uma potência de um número primo. Suponha que $\mu = q^L$, tal que α_i forma uma L-upla sobre GF(q).

A taxa deste código é dada por

$$R = \frac{\log \mu}{N} = \frac{L}{N} \cdot \log(q)$$

O codificador convolucional (N, R, D, M, ν) com $\beta_i = \Gamma(\alpha_i, \alpha_{i-1}, \ldots, \alpha_{i-\nu}, i)$ é tal que

$$\alpha_i \in [GF(q)]^L$$

e

$$\beta_i \in [GF(q)]^N$$

e $\Gamma(\cdot, i)$ são funções lineares dadas por

$$\Gamma(\cdot,\ i)\ :\ [GF(q)]^{(\nu+1)L} \longrightarrow [GF(q)]^{N}$$

Tais funções podem ser representadas por

$$\beta_i = \alpha_i G_o(i) + \alpha_{i-1} G_1(i) + \ldots + \alpha_{i-\nu} G_\nu(i) \qquad (1.1)$$

onde $G_j(i)$ são matrizes L x N sobre GF(q). Para o caso invariante no tempo, temos $\Gamma(\cdot,\ i) = \Gamma(\cdot)$, e assim (1.1) fica dada por

$$\beta_i = \alpha_i G_o + \alpha_{i-1} G_1 + \ldots + \alpha_{i-\nu} G_\nu \qquad (1.2)$$

com G_j matrizes L x N sobre GF(q). O nome *convolucional* provêm de (1.1) ou (1.2) por razões óbvias.

Seria bastante vantajoso ter uma representação matricial para os códigos convolucionais, como ocorre com os códigos de bloco.

Nesta direção, sejam $\alpha_{[u,v]}$ e $\alpha_{[u,v)}$ sequências tais que $\alpha_u, \alpha_{u+1}, \ldots, \alpha_{v-1}, \alpha_v$ e $\alpha_u, \alpha_{u+1}, \ldots, \alpha_{v-1}$. Sejam $\beta_{[u,v]}$ e $\beta_{[u,v)}$ sequências similares à $\alpha_{[u,v]}$ e $\alpha_{[u,v)}$.

Da equação (1.1), temos que

$$\beta_{[0,D+M]} = \alpha_{[0,D)} \begin{bmatrix} G_o(0) & G_1(1) & \cdots & G_\nu(\nu) & & & \\ & G_o(1) & \cdots & G_{\nu-1}(\nu) & G_\nu(\nu+1) & & 0 \\ & & & G_o(\nu) & G_1(\nu+1) & \cdots & \\ & 0 & & & & & \\ & & & & G_o(D-1) & \cdots & G_M(D-1) \end{bmatrix} \tag{1.3}$$

dessa forma a matriz em (1.3) é denominada matriz geradora, G, do código convolucional (N, R, D, M, ν).

Referência

J.L. Massey, "Error Bounds for Tree Codes, Trellis Codes, and Convolutional Codes with Encoding and Decoding Procedures," *CISM Courses and Lectures No. 216*, Viena and New York: Springer-Verlag, 1975.

CAPÍTULO 2

CÓDIGOS CONVOLUCIONAIS SOBRE GF(q): GENERALIZAÇÕES

2.1 - INTRODUÇÃO

Desde a descoberta de códigos convolucionais por Elias, bem como a proposição de um processo de decodificação para tais códigos por Wozencraft-Reiffen tem despertado interesse entre pesquisadores e projetistas de sistemas de comunicações na determinação e/ou proposição de métodos sistemáticos para obtenção desses códigos.

As razões por traz desta busca tem suas bases fundamentadas no fato de que, em geral, os códigos convolucionais apresentam um desempenho bem melhor que os códigos de bloco quando o processo de decodificação emprega o algoritmo de Viterbi ou técnicas de decodificação Sequencial.

Muitos algoritmos foram apresentados e estruturas algébricas foram estabelecidas com o objetivo de tornar sistemático esse processo de codificação. Apesar de todos esses esforços, uma solução geral está para ser proposta.

A dificuldade em se resolver esse problema, ou seja, proposição de um método algébrico ou sistemático será demonstrada no Capítulo 4 através do estabelecimento de um método novo de se caracterizar a geração de códigos convolucionais.

Este método basicamente estabelece algumas regras, matematicamente bem definidas, cujo objetivo é de eliminar um número muito grande de códigos classificados como não eficientes entre todos os possíveis códigos. Em outras palavras, este método utiliza de aspectos de otimização combinatorial relativo ao problema de determinação de códigos convolucionais ótimos quando caracterizamos este problema como o de determinar o fluxo máximo numa rede.

Esta caracterização nos permite relacionar os parâmetros de cada problema da seguinte forma: a representação de estados particionados dos códigos

convolucionais constituirá uma rede de fluxo; os estados nesse diagrama de estados particionados serão os nós da rede; a distância de Hamming de cada transição corresponderá ao fluxo de cada ramo; a propriedade de conservação de fluxo vale, isto é, que a distância total de Hamming saindo ou entrando em cada um dos estados são iguais; o menor valor da soma de todas as distâncias de Hamming pertencentes a um conjunto de corte estão relacionadas com o máximo fluxo através da rede; e finalmente, uma desigualdade relacionando a distância mínima do código convolucional e o máximo fluxo será estabelecida.

2.2 - REVISÃO DE SISTEMAS LINEARES DISCRETOS

Iremos apresentar nesta seção alguns conceitos básicos de sistemas lineares discretos que serão utilizados a seguir. Esses conceitos passam a ser relevantes pois caracterizam o comportamento e estrutura dos códigos de treliça que serão descritos ao longo desses Capítulos.

Inicialmente, considere um sistema linear discreto no tempo descrito pela equação de diferença de estado por

$$E(i+1) = A(i) \cdot E(i) + B(i) \cdot u(i) \qquad (2.1)$$

e equação de saída por

$$T(i) = H(i) \cdot E(i) + D(i) \cdot v(i) \qquad (2.2)$$

onde $i = 0, 1, 2, \ldots$, $E(i)$ é uma matriz coluna composta pelos estados intermediários, $u(i)$ é o controle, $T(i)$ é a saída, $A(i)$ representa a matriz de transição, $B(i)$ é a matriz controle, $H(i)$ é a matriz da condição de saída, $v(i)$ é a

sequência de medida de erro e D(i) é a matriz de medida de erro no instante de tempo t(i) = i. Assumiremos que todos os elementos pertencem aos Reais.

A solução da equação de diferença de estado é dada por

$$E(i) = \chi(i, i_o) \cdot E(i_o) + \sum_{j=i}^{i-1} \chi(i, j+1) \cdot B(j) \cdot u(j), \qquad i \geq i_o + 1 \tag{2.3}$$

onde

$$\chi(i, i_o) = \begin{cases} A(i-1) \cdot A(i-2) \cdot \ldots \cdot A(i_o), & i \geq i_o + 1 \\ I, & i = i_o \end{cases} \tag{2.4}$$

e $E(i_o)$ é a condição de estado inicial.

A solução de estado estacionário, (2.3), é dada por

$$E(i) = \chi(i, i_o) \quad E(i_o) \tag{2.5}$$

Para a condição de sistemas invariantes no tempo, A(i) = A para $i_o \leq i \leq i-1$. Em geral, a matriz A é não diagonal. Como no processo de equivalência dos problemas anteriormente mencionados necessitaremos de uma matriz diagonal, e mais, sabendo que a matriz A é positiva definida, então poderemos determinar uma aplicação tal que a matriz resultante é diagonal.

Seja V uma matriz tal que

$$\chi(i, i_o) = A^{(i-i_o)} = V \cdot \underline{\rho}^{(i-i_o)} \cdot V^{-1} \tag{2.6}$$

$\underline{\rho}$ resulte numa matriz diagonal.

Os elementos ρ_j de $\underline{\rho}$ são os autovalores distintos de A e V é a matriz dos autovetores com v_j o j-ésimo autovetor coluna para $1 < j < n$ e w_j o j-ésimo autovetor de V^{-1}.

Seja $E(i_o) = E_o$, então a solução de (2.5) será

$$E(i) = \sum_{j=1}^{n} \rho_j^{(i-i_o)} \cdot v_j \cdot w_j \cdot E_o$$

No estudo de sistemas lineares é intrínseco o estabelecimento de condições de estabilidade. Não fugindo à esta regra, iremos abordar tal aspecto. A condição de estabilidade está ligada diretamente com os autovalores de matriz A. Assim sendo

Definição 1 : Seja a equação diferença de estado

$$E(i+1) = F[E(i), u(i), i]$$

com condição inicial $E_o(i_o)$. Então a solução nominal estável no sentido de Lyapunov para qualquer $t_o = t_{oi}$ e $\varepsilon \geq 0$, existe um $\delta(\varepsilon, t_o) \geq 0$ tal que $|E(i_o) - E_o(i_o)| \leq \delta$ implica em $|E(i) - E_o(i)| \leq \varepsilon$ para todo $i \geq i_o$.

Como consequência da definição, temos

Teorema 2 : Um sistema linear discreto invariante no tempo $E(i+1) = A \cdot E(i)$

1) é estável no sentido de Lyapunov se e somente se os valores absolutos de

todos os autovalores de A não são maiores do que 1.

2) é assintoticamente estável se e somente se os valores absolutos de todos os autovalores de A são estritamente menores do que 1.

3) é exponencialmente estável se e somente se é assintoticamente estável.

A importância deste Teorema com relação à classificação dos códigos convolucionais se restringe ao fato de que a estabilidade dos mesmos tomados como um sistema linear está diretamente relacionada com a classificação dos códigos serem não catastróficos. Deste modo, se para todo j, $|\rho_j| < 1$, então o código é *não catastrófico*. Por outro lado, se para todo j, $|\rho_j| \geq 1$, então o sistema é instável e o código é dito ser catastrófico.

O conceito de catastrófico implica que existe pelo menos um caminho na treliça com comprimento infinito e a métrica associada finita.

2.3 - PROBLEMA DO AUTOVALOR

Como vimos anteriormente, um código convolucional invariante no tempo pode ser representado pelo diagrama de estado particionado. A evolução temporal do estado pode ser descrita por

$$E(i+1) = A \cdot E(i) + B \qquad (2.8)$$

onde $u(i) = 1$.

A equação de saída é dada por

$$T(i) = H(i) \cdot E(i) \tag{2.9}$$

Resolvendo-se (2.8) e substituindo em (2.9), teremos

$$T = \sum_{k=0}^{\infty} H \cdot A^k \cdot B \tag{2.10}$$

De modo a generalizar este procedimento matemático para códigos convolucionais sobre o corpo de Galois com q elementos, GF(q), onde q é um primo ou uma potência de um primo, definiremos $\pi(\underline{b}, \underline{A}, \underline{h})$ como sendo uma partição da classe de codificadores convolucionais na representação de estados particionados com número de registros igual a ν e taxa $r = b/n$ tal que $\underline{b}$ e $\underline{h}$ são conjuntos de distâncias de Hamming $(q^b - 1)$- dimensionais relativas às condições inicial e de saída, e $\underline{A}$ é um conjunto finito de matrizes de transições correspondentes à cada um dos possíveis arranjos de conexões entre os registros e somadores mod q para cada valor fixado de $\underline{b}$ e $\underline{h}$.

Para cada valor de

$$\underline{b}^t = (b_1, b_2, \ldots, b_n)$$

e

$$\underline{h}^t = (h_1, h_2, \ldots, h_n)$$

onde t significa transposta, pode-se determinar o conjunto de autovalores associados com cada matriz de transição $\underline{A}$. Devemos mencionar que os valores fixos de $\underline{b}$ e $\underline{h}$ estão relacionados com as conexões entre o primeiro e último registros com todos os somadores mod q e que para cada arranjo de conexão entre o primeiro e o último registros permite que se estabeleça a matriz de transição que é

formada pela conexão dos registros restantes com os somadores mod q.

O Teorema de Cayley-Hamilton estabelece que a matriz de transição A satisfaz o polinômio característico P(ρ) dado por

$$P(\rho) = \det(\rho \cdot I - A)$$

onde det(·) significa determinante e que cujas raízes são os autovalores. Uma vez conhecido os autovalores, implica no conhecimento dos autovetores associados. Portanto, podemos determinar as matrizes V e V^{-1} e consequentemente $\underline{\rho}$ a matriz diagonal.

Das equações (2.6) e (2.10), temos

$$T = \sum_{k=0}^{\infty} H \cdot A^k \cdot B = \sum_{k=0}^{\infty} H \cdot V \cdot \underline{\rho}^k \cdot V^{-1} \cdot B$$

onde $\underline{\rho} = \text{diagonal}(\rho_1, \rho_2, \ldots, \rho_n)$.

Seja $\rho^* = \max\{\rho_1, \rho_2, \ldots, \rho_n\}$ e $\underline{\rho}^*$ a correspondente matriz diagonal. Defina $\underline{\rho} < \underline{\rho}^*$ como uma desigualdade termo-a-termo.

Defina $\underline{\alpha}$ e $\underline{\beta}$ como as novas matrizes 1 x (M - 1) e (M - 1) x 1, respectivamente, com M sendo o número de estados tal que

$$\underline{\beta}_{n,1} = \max\{B\} \qquad \text{e} \qquad \underline{\alpha}_{1,n} = \max\{H\}$$

quando pelo menos um elemento de B ou H é zero, ou simplesmente B e H quando todos os elementos são não nulos.

Iremos associar à cada transição no diagrama de estado particionado uma variável independente z^{δ} para todo $\underline{b}$ e $\underline{h}$ não nulos, onde $\delta = w(\underline{b})$ ou $w(\underline{h})$ e

$w(\cdot)$ caracteriza o número de "1" na representação binária de q. Desse modo (2.10) fica representada por

$$T(z) = \sum_{k=0}^{\infty} \underline{\alpha}(z) \cdot A^{k}(z) \cdot \underline{\beta}(z) \qquad (2.11)$$

Sabemos que

$$P_b \leq (1/2 \cdot b) \left| d/dz \right| T(z) \Big|_{z=1}$$

Tomando a derivada de (2.11) teremos

$$P_b \leq (1/2b) \cdot \{\underline{\alpha}'(z) \cdot (I - A(z))^{-1} \cdot \underline{\beta}(z) + \underline{\alpha}(z) \cdot (I - A(z))^{-1} \cdot \underline{\beta}'(z) +$$

$$+ \underline{\alpha}(z) \cdot (I - A(z))^{-1} \cdot A'(z) \cdot (I - A(z))^{-1} \cdot \underline{\beta}(z)\}_{z=1}$$

Como $\underline{\alpha}'(z) = \underline{0}$ e $\underline{\rho} \leq \underline{\rho}^*$, então finalmente teremos

$$P_b \leq (1/2b) \cdot \left\{ \sum_{k=0}^{\infty} \underline{\alpha} \cdot V \cdot \underline{\rho}^* \cdot V^{-1} \cdot \underline{\beta} + \right.$$

$$\left. + \sum_{k=0}^{\infty} \sum_{j=0}^{\infty} \underline{\alpha} \cdot V \cdot \underline{\rho}^* \cdot V^{-1} \cdot V \cdot \underline{\rho}^* \cdot V^{-1} \cdot V \cdot \underline{\rho}^* \cdot V^{-1} \cdot \underline{\beta} \right\}$$

Após algumas operações algébricas chega-se à

$$P_b \leq (1/2b) \cdot \left\{ (D^{h_1+b_1} + D^{h_2+b_2} + \ldots + D^{h_n+b_n}) / (1 - \underline{\rho}^*)^2 \right\} \qquad (2.12)$$

onde D é a função Bhattacharyya e b_i e h_i são os pesos de Hamming das condições iniciais e de saída.

Com isso podemos estabelecer um critério de otimalidade como

Definição 3 : Na classe $\pi(\underline{b}, \underline{A}, \underline{h})$ de codificadores convolucionais não catastróficos (assintoticamente estáveis como sistema) sobre GF(q) com um número de registros K, memória $\nu = K - 1$, e taxa $r = b/n$, um codificador é dito ser *ótimo* se e somente se o mesmo faz com que o menor valor do limitante superior da probabilidade de erro de bit dada por (2.12) para valores fixos de $\underline{b}$ e $\underline{h}$ seja alcançado.

Como consequência desta definição temos

Lema 4 : Para valores fixos de $\underline{b}$ e $\underline{h}$ na classe $\pi(\underline{b}, \underline{A}, \underline{h})$ de codificadores convolucionais não catastróficos sobre GF(q) com memória ν e taxa $r = b/n$ e $\gcd(b, n) = 1$, o codificador *ótimo* é aquele que possui o menor dos ρ^* entre todos os máximos valores dos autovalores associados com o conjunto $\underline{A}$.

Demonstração : Desenvolvimento até a equação (2.12). CQD

Lema 5 : Na classe $\pi(\underline{b}, \underline{A}, \underline{h})$ de codificadores convolucionais não catastróficos sobre GF(q) com memória ν e taxa $r = b/n$, para $\pi(\underline{b}_1, \underline{A}_1, \underline{h}_1)$ e $\pi(\underline{b}_2, \underline{A}_2, \underline{h}_2)$ tal que $\underline{b}_1 \neq \underline{b}_2$ e $\underline{h}_1 \neq \underline{h}_2$, termo-a-termo; se $\underline{\rho}_1 = \underline{\rho}_2$ são os valores máximos dos autovalores associados a $\underline{A}_1$ e $\underline{A}_2$, respectivamente, então o codificador ótimo é aquele tal que $\underline{b}_i + \underline{h}_i$, $i = 1, 2$, é o maior.

Demonstração : Substituição das hipóteses em (2.12). CQD

Dos Lemas 4 e 5, chega-se que P_b atinge a igualdade quando ρ^* é o menor dos máximos valores dos autovalores e $\underline{b} + \underline{h}$ é o maior. Deste modo, quando ρ^* é fixo, a taxa é r = 1/n e o corpo é o GF(2), P_b assume o menor limitante superior quando $b_1 = h_1 = n$. Isto implica que o peso de Hamming deste vetor binário é n. Por outro lado, no GF(q) se os vetores $\underline{b}$ e $\underline{h}$ sendo (q - 1)-dimensionais são (quase) igualmente distribuídos os termos $b_i + h_i$ são (quase) iguais e o maior possível. Por outro lado, para taxas r = b/n e o corpo GF(q) cada um dos vetores $(q^b - 1)$-dimensionais representam os pesos de Hamming de uma sequência de saída associada, pode-se mostrar que a soma desses pesos de Hamming é $n \cdot (q-1) \cdot q^{b-1}$. Como o peso total de Hamming está associado ao fluxo entrando ou saindo dos estados, temos que este valor é o fluxo entre transições na rede.

Seja Γ o conjunto de corte, isto é, o conjunto formado por todos os cortes possíveis que irão desconectar a fonte do destino. Seja γ_i o valor associado ao i-ésimo corte de Γ. Seja $\gamma_* = \min_i\{\gamma_i\}$. Então γ_* é o valor do fluxo máximo de uma rede.

Como será demonstrado adiante, $n \cdot (q-1) \cdot q^{b-1} = \gamma_*$ e portanto este é o máximo fluxo. Consequentemente, a propriedade de máximo fluxo está relacionada com o menor valor do autovalor entre os máximos autovalores de $\underline{A}$.

Resta-nos relacionar a propriedade de conservação de fluxo com o menor valor do autovalor entre os máximos autovalores de $\underline{A}$. Para tal, usaremos da propriedade de matrizes aumentadas.

Seja $\tilde{A}$ a matriz de transição aumentada, isto é, incluiremos em $\tilde{A}$ os estados iniciais e de saída. Para valores de r = b/n e $(\nu + 1)$ e tendo em mente

que estamos em GF(q), $\tilde{A}$ é uma matriz $(M+1) \times (M+1)$ onde $M = q^{b.\nu}$. Do Teorema de Cayley-Hamilton o polinômio característico de A é

$$P(\rho) = \det(\rho \cdot I - A) = \underline{\rho}^2 \cdot \det(A)$$

de tal forma que $\tilde{A}$ contem os mesmos autovalores que A.

Defina $L_i(\tilde{A})$ como o produto dos elementos não nulos da i-ésima linha de $\tilde{A}$ e $C_j(\tilde{A})$ como o produto dos elementos não nulos da j-ésima coluna de $\tilde{A}$, para todo i e j, isto é,

$$L_i(\tilde{A}) = \prod_{j=1}^{n} \tilde{a}_{i,j} \qquad \text{e} \qquad C_j(\tilde{A}) = \prod_{i=1}^{n} \tilde{a}_{i,j}$$

Dada a dificuldade de se provar que *sempre* existe um código ótimo satisfazendo as propriedades de conservação de fluxo e máximo fluxo, embora todos os códigos ótimos encontrados satisfaçam tais propriedades, a seguir estabeleceremos uma conjectura.

Conjectura 6 : Se existe um codificador convolucional invariante no tempo e não sistemático com parâmetros ν e $r = b/n$ tal que $L_i(\tilde{A}) = C_j(\tilde{A}) = D^{\Phi}$ para todo i e j com $\Phi = n \cdot (q-1) \cdot q^{b-1}$ e ρ^* o menor autovalor entre os máximos autovalores de $\tilde{A}$, então o código é *ótimo*.

2.4 - CÓDIGOS CONVOLUCIONAIS COMO UM PROBLEMA DE FLUXO EM REDE

Como mencionado na seção 2.1, o diagrama de estado particionado cons

tituirá uma rede ou um grafo direcionado.

Seja um grafo direcionado matematicamente descrito por $G = (N, R)$ onde N é um conjunto finito de elementos denominados nós ou estados, isto é, $N = \{1, 2, \ldots, n\}$ e onde R é um conjunto também finito de pares ordenados (i, j) denominados ramos, isto é, $R = \{(i, j) : i, j \in N\}$.

Seja $c_{i,j}$ um número não negativo associado ao ramo (i, j). Denominamos $c_{i,j}$ a capacidade do ramo (i, j). Do conjunto N, os elementos 1 e n serão designados a origem e destino, respectivamente.

Seja $x_{i,j}$, $x_{i,j} \geq 0$, o fluxo dos ramos (i, j), para todo i e j, do grafo direcionado G. A descrição matemática dos fluxos numa rede é dada pela equação (2.13)

$$\sum_i x_{i,j} - \sum_k x_{j,k} = \begin{cases} -\Phi & \text{se } j = 1 \\ 0 & \text{se } j \neq 1, n \\ \Phi & \text{se } j = n \end{cases} \qquad (2.13)$$

sujeito à restrição $0 \leq x_{i,j} \leq c_{i,j}$.

A interpretação da equação (2.13) segue: se $j = 1$, então a primeira somatória é nula pois não existem fluxos de ramos chegando no nó 1 e a segunda somatória fornece a soma de todos os fluxos de ramo saindo do nó 1. Se $j = n$, então só teremos a primeira somatória visto que a segunda fornece a soma de todos os fluxos de ramos saindo do destino. Por outro lado, se $j \neq 1, n$, então ambas somatórias resultam no mesmo valor de fluxo total saindo e/ou entrando em cada nó da rede. Quando isto acontece, diz-se que a rede é limitada e balanceada.

A afirmação feita acima para os casos $j = 1$ e $j = n$ de que a soma de

todos os fluxos de ramos iguala ao máximo fluxo da rede está baseada no Teorema do Máximo-Fluxo Mínimo-Corte que afirma que o fluxo máximo em uma rede é igual à soma dos fluxos dos ramos pertencentes ao mínimo conjunto de corte. Por outro lado, se a conjectura 6 da seção anterior for verdadeira então a equação (2.13) é válida para os códigos convolucionais. Consequentemente, a determinação de códigos convolucionais ótimos não sistemáticos e específicos pode ser caracterizada como um problema de fluxo máximo em redes.

Assim o nosso primeiro passo na solução desse problema está em determinarmos explicitamente o fluxo máximo Φ que deverá ser satisfeito para cada uma das classes de códigos convolucionais.

Teorema 7 : O fluxo máximo Φ para códigos convolucionais não sistemáticos, específicos e invariantes no tempo sobre GF(q) com memória ν e taxa $r = b/n$ é dado por

$$\Phi = n \cdot (q-1) \cdot q^{b-1}$$

Demonstração : Para taxa $r = b/n$ e memória ν o número de estados é $q^{b.\nu}$. O número de transições de cada estado é q^b. Assim o número total de ramos é $q^{b.(\nu+1)}$. Por outro lado, como a saída codificada tem um comprimento n, então teremos q^n possíveis sequências de saída.

Seja α a razão entre o número de ramos e o número de possíveis sequências de saída.

Se $\alpha > 1$, então este número especificará quantas vezes as sequências de saída se repetirão. Se $\alpha \leq 1$, então ela fornecerá a proporção das sequências de saída que serão utilizadas.

Seja $w_T(C)$ o peso total de Hamming das sequências de saída, isto é,

$$w_T(C) = \sum_{m=0}^{n} m \cdot (q-1)^m \cdot C_{n,m} =$$

$$= (q-1) \cdot [d/dt]\ (1+a)^n \Big|_{a=q-1}$$

$$= n \cdot (q-1) \cdot q^{n-1}$$

Então o produto $\alpha \cdot w_T(C)$ fornecerá o peso total de Hamming com relação a rede. Como existem $q^{b.\nu}$ estados, o fluxo uniforme será

$$\Phi = \alpha \cdot w_T(C)/q^{b.\nu} = n \cdot (q-1) \cdot q^{b-1} \qquad \text{CQD}$$

Corolário 8 : As palavras de ramo do código que entram e saem qualquer nó da rede são complementares quando $b = 1$ e $q = 2$.

Dessa forma a redução do problema original passa a ser: Seja dada uma rede onde todos os fluxos satisfazem $0 \leqslant w_{i,j} \leqslant c_{i,j}$, onde $w_{i,j}$ é a distância de Hamming do ramo (i, j)

Problema I

$$\max_{A} \Big\{ \sum_{j \in A} c_j \cdot w_{i,j} \Big\}$$

sujeito à

$$\sum_{i} w_{i,j} - \sum_{k} w_{j,k} = \begin{cases} -\Phi, & \text{se} \quad j = 1 \\ 0, & \text{se} \quad j \neq 1,\ n \\ \Phi, & \text{se} \quad j = n \end{cases}$$

onde A é o conjunto de nós intermediários e

$$c_j = \begin{cases} 1, & \text{se } j \in A \\ 0, & \text{se } j \notin A \end{cases}$$

Esta redução é característica de problemas combinatoriais que sob certas condições podem ser facilmente solucionados. No caso mais geral, este problema combinatorial não é fácil de ser resolvido, porém, podemos aplicar o processo de solução através do estabelecimento de alguns parâmetros que excluirão numa primeira aproximação um número grande de soluções factíveis. Um desses parâmetros é a distância mínima do código.

Dessa forma teremos

Teorema 9 : Para qualquer código convolucional, invariante no tempo, periodicamente variante no tempo, variante no tempo e códigos de treliça não lineares sobre GF(q) com memória ν, taxa $r = b/n$ e $\gcd(b, n) = 1$, a distância mínima é limitada superiormente por

$$d_{min} \leq \min_{p \geq 1} \{(q-1) \cdot [q^{p-1}/(q^p - 1)] \cdot (n/b) \cdot (p + b \cdot \nu)\}$$

Demonstração : É de conhecimento geral que um código convolucional q-ário terminado com M símbolos de informação é um código de grupo cuja matriz geradora G tem dimensão $M \times n \cdot (M + \nu)$ para taxas $r = 1/n$. Para taxas $r = b/n$, $\gcd(b, n) = 1$ como teremos b entradas paralelas com memória ν, de modo a terminar o código

teremos que inserir $b \cdot \nu$ dígitos conhecidos a priori.

O comprimento total dos bits de informação é $b \cdot M$ e então a matriz geradora terá agora $b \cdot M$ linhas e $(n/b) \cdot (b \cdot M + b \cdot \nu)$ colunas.

Para códigos de grupo q-ário, a distância de Hamming entre palavras do código é equivalente ao peso das palavras não nulas. Assim, se todas as $q^{b.M}$ palavras forem arranjadas como linhas de uma matriz, o peso total dessas $q^{b.M}$ palavras é limitado superiormente por

$$\tilde{w} \leq (q-1) \cdot q^{b.M-1} \cdot (n/b) \cdot (b \cdot M + b \cdot \nu)$$

Como $q^{b.M} - 1$ palavras são não nulas e seu peso mínimo deve ser menor ou igual ao peso médio, a distância mínima satisfaz

$$d_{min} \leq (q-1) \cdot [q^{b.M-1}/(q^{b.M}-1) \cdot (n/b) \cdot (b \cdot M + b \cdot \nu)] = \gamma(b \cdot M)$$

Como este superior vale para todo $b \cdot M$, também valerá para $p < b \cdot M$. Note que estamos interessados no menor valor de $\gamma(b \cdot M)$. Deste modo, o próximo passo é então

$$\gamma(p^*) = \min_{p}\{\gamma(p)\}$$

onde p^* é a solução desejada. Esta minimização será feita sob a hipótese de que $p \in R$. Isto é um abuso matemático com relação ao verdadeiro domínio da variável p que pertence ao Z^+. De qualquer forma, estaremos utilizando deste procedimento sendo que o valor final de p será convertido ao conjunto dos inteiros positivos. Desse modo teremos

$$d_{min} \leq [\gamma(p^*)]$$

onde $[a]$ significa o maior inteiro menor ou igual a a. CQD

Agora, para que o Problema I seja realmente uma redução é necessário

que exista uma conexão entre o máximo fluxo Φ e a distância mínima, d_{min}. Esta conexão será viabilizada através do seguinte

Teorema 10 : Para qualquer código convolucional sobre o GF(q) com memória ν, taxa $r = b/n$ e $\gcd(b, n) = 1$, a distância mínima é limitada superiormente por

Caso 1 : $b = 1$ e $q = 2$ (conjunto de corte unitário)

$$d_{min} \leq \alpha_1 \cdot \Phi$$

onde

$$\alpha_1 = \left[2^{p^*-1}/(2^{p^*} - 1)\right] . (p^* + \nu)$$

Caso 2 : a) $b \geq 2$ e q um primo ou potência de um primo; b) $b = 1$ e q um primo ou potência de um primo com $q \neq 2$ (conjunto de corte arbitrário)

$$d_{min} \leq \alpha_2 \cdot \Phi$$

onde

$$\alpha_2 = \left[q^{p^*}/(q^{p^*} - 1\right] . \nu$$

Demonstração : Sabemos que $\Phi = n . (q - 1) . q^{b-1}$

Caso 1 : Para $b = 1$ e $q = 2$, conjunto de corte unitário, temos que $2 . b = 2$ transições de cada nó, menos do nó 1 (origem). Se o conjunto de corte contém a transição do nó 1 para o nó 2 ou a transição do nó $(\tilde{n} - 1)$ para o nó $\tilde{n}$,

$$\max \Phi = \max w_{1,2} = \max w_{\tilde{n}-1,\tilde{n}}$$

Desse modo, $\max w_{1,2} = \max w_{\tilde{n}-1,\tilde{n}} = n$, uma vez que o peso de Hamming máximo deste ramo de transições vale n. Por outro lado, para qualquer conjunto de corte com cardinalidade maior do que 1, temos que este valor é pelo menos Φ, pois

é suficiente levar em consideração o primeiro nó (logo após a transição do nó 1) ou o último nó, $\tilde{n}-1$ (logo antes da transição que conduzirá ao nó $\tilde{n}$). Em qualquer um dos casos, as transições saindo do nó 1 ou as transições indo para o nó $\tilde{n}$ tem peso de Hamming igual a Φ, uma vez que a conservação de fluxo vale. Portanto, substituindo $b=1$ e $n=\Phi$ no Teorema 9 concluimos a demonstração.

Caso 2 : $b \geq 2$ e q um primo ou uma potência de um primo, $b=1$ e q um primo ou potência de um primo com $q \neq 2$ (conjunto de corte arbitrário). Sabemos que

$$\Phi = n \cdot (q-1) \cdot q^{b-1} = \sum_{i,j \in CC} w_{i,j}$$

onde CC = conjunto de corte.

Seja $p = p^*$ a solução ótima de $\gamma(p)$, isto é, $d_{min} \leq \gamma(p^*)$. Assim,

$$d_{min} \leq \left\{(q-1) \cdot \left[q^{p^*-1}/(q^{p^*}-1)\right] \cdot (n/b) \cdot (p^* + b \cdot \nu)\right\}$$

$$= \left\{(q-1) \cdot \left[q^{p^*-1}/(q^{p^*}-1)\right] \cdot (n/b) \cdot \nu \cdot (p^*/\nu + b)\right\}$$

Multiplicando a equação acima por ln q no numerador e no denominador e rearranjando os termos, teremos

$$d_{min} \leq \left\{(q-1) \cdot \left[q^{p^*-1}/(q^{p^*}-1)\right] \cdot Z\right\}$$

onde

$$Z = (n/b \cdot \ln q) \cdot \left[(p^* \cdot \ln q)/\nu + b \cdot \ln q\right]$$

Como $(p^* \cdot \ln q)/\nu \leq \ln q$, iremos limitar superiormente esta quantidade por

$$(p^* \cdot \ln q)/\nu \leq \sum_{n=1}^{\infty} (b \cdot \ln q)^n/(n-1)!$$

$$= (b \cdot \ln q) \cdot (q^b - 1)$$

Substituindo este limitante em d_{min}, teremos

$$d_{min} \leq \left\{(q-1) \cdot [q^{p^*-1}/(q^{p^*}-1)] \cdot (n/b \cdot \ln q) \cdot \nu \cdot ([b \cdot \ln q] \cdot (q^b - 1) + b \cdot \ln q)\right\}$$

$$= \left\{n \cdot (q-1) \cdot (q^b - 1) \cdot [q^{p^*}/(q^{p^*}-1)] \cdot \nu\right\}$$

$$= \alpha_2 \cdot \Phi$$

Vale a pena reforçar que o Teorema acima estabelece a solução do Problema I, consequentemente que a busca do(s) código(s) ótimo(s) deve iniciar entre os códigos não sistemáticos.

O exemplo a seguir demonstra o procedimento até aqui.

Exemplo : Determine o código convolucional ótimo sobre o GF(2) quando $q = 2$, $\nu = 2$ e $r = 1/85$. O fluxo máximo é $\Phi = n \cdot 2^{b-1} = 85$. Assim, os ramos saindo e entrando nos nós 1 e n tem peso de Hamming 85, isto é, o primeiro e último registros estão conectados a todos os ou-exclusivos. A distância mínima $d_{min} \leq 226$. Se a igualdade deve prevalecer, então o registro intermediário deverá estar conectado a 56 ou-exclusivos dentre os 85. Qualquer uma das possíveis conexões resulta em códigos equivalentes. Os demais pesos de Hamming seguem da conservação de fluxo.

2.5 - CRITÉRIO DE SELEÇÃO PARA OS CÓDIGOS ÓTIMOS

Como vimos nas seções anteriores, existe um codificador ótimo para todas as partições $\pi(\underline{b}, \underline{A}, \underline{h})$. Também foi visto que para a partição $\pi(n, \underline{A}, n)$ o limitante superior da probabilidade de erro de bit é o menor quando ρ^* é se-

lecionado como sendo o menor entre os maiores valores dos autovalores. Quando isso acontece temos que a propriedade de conservação de fluxo é válida em todos os nós intermediários da rede.

Embora as propriedades de conservação e máximo fluxo sejam importantes, elas somente delimitam a busca do código ótimo a um conjunto cuja cardinalidade é menor que a do problema original. Portanto, os elementos indispensáveis no selecionamento de códigos ótimos são a conservação e máximo fluxo e a distância mínima do código.

Seja χ o diagrama de estado particionado de um código convolucional sobre GF(q) com r = b/n e memória ν. Seja $\underline{d}$ o conjunto ordenado das distâncias de Hamming acumuladas ao longo dos $\nu+1$ ramos que saem do nó 1 e terminam no nó n. Esse conjunto é explicitamente dado por

$$\underline{d} = \left\{ d_1, d_2, \ldots, d_{q^b-1} \right\}$$

onde $d_i = w_{1i} + w_{in}$ para $\nu = 1$ e w_{1i} e w_{in} são os pesos de Hamming que saem do nó 1 e entram no nó n, respectivamente. É natural que

$$d_\infty \geq \min \{ \underline{d} \}$$

onde d_∞ é a distância livre do código.

O critério de selecionamento adotado é do tipo lexicográfico, isto é, se X e Y são dois conjuntos com n elementos, diremos que $X > Y$ se $X_i = Y_i$ para $1 \leq i \leq k$ e $X_{k+1} > Y_{k+1}$.

Devemos ressaltar que este critério não implica que o código seja ótimo em termos da probabilidade de erro de bit. Por outro lado, sabemos que

$\Phi = n \cdot (q-1) \cdot q^{b-1}$ é a soma de todas as distâncias de Hamming que saem ou entram em qualquer nó da rede. Como o codificador possue $\nu+1$ registros então $\Phi \cdot (\nu+1)$ é um limitante superior para a soma dos d_i pertencentes aos caminhos com $(\nu+1)$ ramos. Mas isto é equivalente à um problema de programação linear estabelecido da seguinte forma: sejam dados os valores de b, n e ν, então

Problema II

Determine a B-upla $\hat{a}_i = (a_{1i}, a_{2i}, \ldots, a_{Bi})$ lexicograficamente ótima de

$$\sum_{i=1}^{B} a_i \cdot d_i \leqslant (\nu+1) \cdot \Phi \qquad (2.14)$$

sujeito a condição de conservação de fluxo e que $d_1 = d_\infty$, $d_2 = d_\infty + 1$, etc.

Note que (2.14) possue no máximo q^b possíveis soluções e que as mesmas estarão sendo comparadas lexicograficamente.

Uma vez que o valor de $\hat{a}_i$ é determinado, o próximo passo será a solução do seguinte problema

Problema III

Determine as $(\nu+1)$ submatrizes geradoras G_i com elementos em GF(q) tal que a combinação linear das linhas de G_i satisfaçam a solução do Problema II.

Como um exemplo de demonstração do critério de seleção, considere o caso onde $\nu = 1$ e $r = 2/11$. Do Teorema $\Phi = 11 \cdot 2 = 22$. Do Teorema, $d_{min} \leq 14$,

então o Problema II é estabelecido da seguinte forma:

$$a_1 \cdot d_1 + a_2 \cdot d_2 + a_3 \cdot d_3 \leq 44$$

com $d_1 = 14$, $d_2 = 15$ e $d_3 = 16$. Então

$$14 \cdot a_1 + 15 \cdot a_2 + 16 \cdot a_3 \leq 44 \qquad (2.15)$$

Os conjuntos de soluções de (2.15) são

$$\hat{a}_1 = (1, 2, 0) \qquad \text{e} \qquad \hat{a}_2 = (2, 0, 1)$$

Este resultado nos diz que $\underline{d} = \{14, 15, 15\}$ para $\hat{a}_1$ e $\underline{d} = \{14, 14, 16\}$ para $\hat{a}_2$. Do critério de seleção (lexicográfico) temos que o conjunto $\{14, 15, 15\}$ é lexicograficamente maior que $\{14, 14, 16\}$. Portanto, $\underline{d} = \{14, 15, 15\}$.

O Problema III, inerentemente combinatorial, consiste da determinação das sub matrizes G_o e G_1 tal que 14, 15, 15. é verificado. Mas os d_i's são resultados da soma dos pesos de Hamming w_{1i} e w_{in}, isto é,

$$d_i = w_{1i} + w_{in}, \qquad \text{para} \quad 1 \leq i \leq n$$

Uma possível solução deste problema combinatorial é

$$w_{11} = 7,\ w_{12} = 8,\ w_{13} = 7 \quad \text{e} \quad w_{1n} = 7,\ w_{2n} = 7,\ w_{3n} = 8$$

e as correspondentes sub matrizes são

$$G_o = \begin{bmatrix} 1\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1\ 1 \\ \\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1 \end{bmatrix} \qquad G_1 = \begin{bmatrix} 0\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1 \\ \\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0 \end{bmatrix}$$

Note que a solução do Problema II forneceu o número de palavras do código com as distâncias de Hamming d_1, d_2 e d_3. Por outro lado, a solução do Problema III forneceu as sub matrizes geradoras do código que realmente geram as palavras do código com os pesos especificados e com o número fornecido pela solução do Problema II.

Se isto não tivesse ocorrido, teríamos que resolver novamente o Problema II com a alteração dos valores de d_i para $d_i - 1$, e novamente resolver o Problema III. Se os resultados não forem iguais, voltaríamos a alterar os últimos valores de d_i de uma unidade. Esse procedimento só irá terminar quando for possível encontrar sub matrizes que gerem as palavras do código com os pesos especificados no Problema II.

2.6 - ALGORITMO DE SELECIONAMENTO

Das seções anteriores podemos concluir que das propriedades de conservação de fluxo e máximo fluxo o limitante superior da P_b atinge seu menor valor. A consequência imediata deste fato, faz com que a cardinalidade do conjunto original seja reduzida sensivelmente de $q^{b.n.(\nu+1)}$ para $q^{b.n.\nu}$. Somente esta redução não é suficiente para alcançar o objetivo de busca num conjunto com cardinalidade pequena. O novo passo nessa direção utiliza da solução do Problema II conjuntamente com a do Problema III. Isso faz com que a cardinalidade do conjunto resultante da solução dos Problemas II e III seja q^b como bem pode

ser visto do Problema II e III. Ainda assim a cardinalidade é grande dependendo dos valores de q e b obviamente, porém, é a menor que se consegue com a redução do problema de determinação do máximo fluxo numa rede. A proposição de um algoritmo que resolvesse esse problema em tempo polinomial seria um resultado fantástico.

O procedimento de busca exaustiva dentro do conjunto resultante da solução conjunta dos Problemas II e III passa a ser o modus operandi.

Este procedimento de busca utiliza dos seguintes elementos: função enumeradora T(z) descrita por um polinômio em z tendo como coeficientes o parâmetro D^{δ} onde D é a função de Bhattacharyya e δ o peso de Hamming acumulado associado à um caminho na rede. O grau do polinômio caracteriza o erro de decodificação, ou seja, o número de bits de informação em erro.

Em geral T(z) é dada por

$$T(z) = \gamma_1 \cdot D^{\alpha_1} \cdot z^{\beta_1} + \gamma_2 \cdot D^{\alpha_2} \cdot z^{\beta_2} + \ldots$$

onde γ_i, α_i, β_i, para $i \geq 1$ são constantes inteiras.

Seja Δ uma árvore q^b-ária assim constituída: o número de ramos saindo da raiz é $(q^b - 1)$ e para cada um desses nós saem q^b ramos até uma profundidade $\nu + 1$ com ν podendo tender a infinito.

Para facilitar o entendimento e descrição do algoritmo vamos supor que $q = 2$ e que esse árvore tenha profundidade $\nu + 1$. Desta forma teremos uma árvore binária com a primeira transição da raiz com $(2^b - 1)$ elementos e $\nu + 1$ ramos de transições da raiz até o nó terminal.

Esta árvore caracteriza eventos de erro quando assumimos que a se-

quência nula for a transmitida. Deste modo, se houver uma transição na árvore binária isto acontecerá devido à um erro de decodificação. Todos os ramos que seguem a esta primeira transição estão associados à seguinte regra de convenção: decodificação correta de bits de informação "0" move para cima e decodificação erronea de bits de informação "1" move para baixo. Sempre que houver uma decodificação erronea a variável z aparecerá no polinômio.

Seguindo esta convenção e sabendo que P_b está relacionada com a derivada de T(z) com relação à z, a *idéia chave* do algoritmo de busca é conjuntamente de minimizar os coeficientes de $[d/dz]\ T(z)$ e maximizar os expoentes de D.

Em geral, procura-se fixar $a_1 = d_\infty$ para o caminho mais curto na árvore de decodificação quando ν é par e $a_1 = d_\infty + 1$ quando ν é impar. Devemos salientar ainda que a_1 representa o número de conexões entre os ou-exclusivos e os registros. Através de uma escolha criteriosa, embora heurística, distribue-se a distância de Hamming ao longo do caminho mais curto de forma que esta soma iguale o valor de a_1 e preservando a propriedade de máximo fluxo. A consequência desse passo é que uma conexão tentativa entre ou-exclusivos e registros fica estabelecida.

O passo seguinte é o determinar os fluxos $a_{\nu+2+j}$ dos ramos ao longo das $(\nu + 2 + j)$ transições. Para tal temos:

i) Inicialmente fixe $j = 0$

ii) Para $\nu + 1$ impar [par], se $a_{\nu+2+j} < a_{\nu+1+j}$ [vá para iv)], rearranje as conexões até que pelo menos $a_{\nu+2+j} = a_{\nu+1+j}$

iii) Se $a_{\nu+2+j} \geq a_{\nu+1+j}$, determine os fluxos para os $(\nu + 3 + j)$ ramos ao longo do caminho

iv) Compare $a_{\nu+3+j}$ com $a_{\nu+2+j}$. Se for menor, rearranje as conexões tal que os valores anteriores não violem a *idéia chave*. Se maior ou igual, faça $j = j + 1$ e siga para iv). Repita este procedimento até que todos os ramos da árvore de decodificação com profundidade $\nu + 1$ sejam determinados.

Este algoritmo possibilita que uma busca rápida seja realizada e como resultado teremos: o polinômio enumerador do código bem como as conexões dos ou-exclusivos e registros.

A seguir apresentaremos as Tabelas dos códigos ótimos para uma gama variada de memórias e taxas.

Tabela 1

Memória $\nu = 2$, GF(2)

taxa	d_∞	função geradora (octal)	
1/3n	8.n	$5^n \quad 7^{2.n}$	$n > 0$
1/(3.n+1)	8.n+2	$5^{n+1} \quad 7^{2.n}$	$n > 0$
1/(3.n+2)	8.n+5	$5^{n+1} \quad 7^{2.n+1}$	$n \geqslant 0$

Tabela 2

Memória $\nu = 3$, GF(2)

taxa	d_∞	função geradora (octal)	
1/3n	10.n	$13^n \quad 15^n \quad 17^n$	$n > 0$
1/(3.n+1)	10.n+3	$13^n \quad 15^{n+1} \quad 17^n$	$n > 0$
1/(3.n+2)	10.n+6	$13^n \quad 15^{n+1} \quad 17^{n+1}$	$n \geqslant 0$

onde x^a significa que o valor x aparece a vezes.

Tabela 3

Memória $\nu = 4$, GF(2)

taxa	d_∞	função geradora (octal)
1/9	36	25 25 27 33 33 35 35 37 37
1/10	40	25 25 25 33 33 33 35 37 37 37
1/11	44	25 25 25 27 33 33 33 35 37 37 37
1/12	48	25 25 25 27 33 33 33 35 35 37 37 37
1/13	52	25 25 25 27 27 33 33 33 35 35 37 37 37
1/14	56	25 25 25 27 27 33 33 33 35 35 35 37 37 37
1/15	60	25 25 25 25 27 33 33 33 33 35 35 37 37 37 37
1/16	64	25 25 25 25 27 33 33 33 33 35 35 35 37 37 37 37

Tabela 4

Memória $\nu = 5$, GF(2)

taxa	d_∞	função geradora (octal)
1/9	41	51 57 57 65 65 67 71 73 77
1/10	45	45 47 53 55 65 67 73 73 75 77
1/11	50	47 53 57 57 65 65 65 71 73 75 77
1/12	54	51 53 55 57 65 65 65 67 71 73 77 77
1/13	59	47 51 53 53 57 65 65 67 67 73 75 75 77
1/14	63	47 51 53 57 57 63 65 65 65 67 73 75 75 77
1/15	68	47 51 53 57 57 63 65 65 65 67 73 75 75 75 77
1/16	72	47 53 57 57 63 65 65 65 67 67 71 71 73 75 75 75

Tabela 5

Memória $\nu = 6$, GF(2)

taxa	d_∞	função geradora (octal)
1/9	46	117 123 127 137 153 155 165 171 175
1/10	51	115 115 127 133 137 145 157 165 173 175
1/11	56	115 115 125 127 137 151 157 163 173 173 175
1/12	61	115 117 125 125 133 135 153 157 167 171 171 175
1/13	66	115 117 125 127 133 135 153 155 157 167 171 171 175
1/14	72	115 117 125 127 133 135 153 155 157 167 171 171 175 175
1/15	76	117 127 127 131 131 135 135 153 153 155 157 171 171 175 175
1/16	82	117 127 127 131 131 135 135 137 153 153 155 157 171 171 175 175

Tabela 6

Memória $\nu = 7$, GF(2)

taxa	d_∞	função geradora (octal)
1/9	51	251 265 267 273 311 337 337 337 345
1/10	56	225 225 273 275 275 313 317 357 363 365
1/11	62	225 257 267 277 277 315 327 331 351 355 363
1/12	68	225 257 267 277 277 315 327 331 345 351 355 363
1/13	74	225 257 257 267 277 277 315 327 331 345 351 355 363
1/14	80	225 257 257 267 267 277 277 315 327 331 345 351 355 363
1/15	85	225 235 257 257 267 267 277 277 315 327 331 345 351 355 363
1/16	91	231 255 257 257 267 267 275 277 277 315 327 331 345 351 355 363

Tabela 7

Memória $\nu = 8$, GF(2)

taxa	d_∞	função geradora (octal)
1/5	31	467 531 535 675 747
1/6	37	475 545 553 677 711 727
1/7	44	457 463 525 673 737 751 755
1/8	50	513 553 567 625 647 671 717 775
1/9	56	471 515 527 537 653 661 673 747 775
1/10	62	467 537 547 571 625 653 677 711 725 773

Tabela 8

Memória $\nu = 1$, GF(2)

taxa	d_∞	função geradora (octal)	
2/9	12	733 547	714 473
2/10	13	1760 0177	0774 1037
2/11	14	3163 1755	1077 3760
2/12	16	7760 0377	1774 3761
3/9	10	760 076 651	370 742 147
3/10	11	1740 0374 0547	0770 0037 1313
3/11	12	3434 0770 2133	2525 2752 1361
3/12	13	7740 7036 3147	4614 7660 0367
4/8	8	160 074 063 252	265 232 170 116
4/9	8	740 146 264 453	170 147 716 444
4/11	11	3740 0374 0707 1625	3700 0370 0516 1225

Tabela 9

Memória $\nu = 2$, GF(2)

taxa	d_∞	função geradora (octal)								
2/9	18	733	547		714	473		157	723	
2/11	22	3163	1755		1077	3760		3617	3163	
3/5	7	23	35	22	25	06	16	25	31	36
3/7	12	131	036	063	125	071	162	056	033	126
3/8	13	360	074	227	146	073	303	370	066	033
3/10	16	1740	0374	0547	0770	0037	1313	0525	0553	1660
3/11	18	3434	0770	2133	2525	2752	1361	3314	0476	2547

Tabela 10

Memória $\nu = 3$, GF(2)

taxa	d_∞	função geradora (octal)											
2/9	23	732	547		714	477		167	732		473	335	
2/11	28	3163	1755		1137	1762		1656	3563		2746	1437	
3/5	10	23	35	24	15	06	32	25	31	36	21	12	34
3/7	14	130	036	063	127	071	162	057	033	147	114	121	052
3/8	17	344	076	227	146	073	303	330	063	033	313	125	346
3/10	22	1740	0370	1167	0770	0037	1313	0525	0553	1662	1272	0317	0546
3/11	24	1524	0372	3113	2275	1646	3115	1742	0672	2457	1343	2336	1534

Tabela 11

Memória $\nu = 4$, GF(2)

taxa	d_∞	função geradora (octal)									
2/5	14	32	15	13	37	34	22	15	31	26	13
2/7	21	113	076	073	055	136	145	131	156	174	073
2/9	27	536	647	554	333	374	672	433	735	762	175
2/11	34	3056	1771	3227	1555	0337	2742	3456	3563	3346	2437

Referências

P. Elias, "Coding for Noisy Channels," *IRE Conv. Rec.*, Part 4, pp. 37-47, 1955.

J.M. Wozencraft, e B. Reiffen, *Sequential Decoding*, MIT Press, Cambridge, Mass., 1961.

G.D. Forney, "Convolutional Codes I: Algebraic Structure," *IEEE Trans. Inform. Theory*, Vol. IT-16, pp. 720-738, November 1970.

J.P. Odenwalder, "Optimal Decoding of Convolutional Codes," Ph.D. Dissertation, University of California, Los Angeles, 1970.

K.J. Larsen, "Short Convolutional Codes with Maximum Free Distance for Rates 1/2, 1/3, 1/4," *IEEE Trans. Inform. Theory*, Vol. IT-19, pp. 371-372, May 1973.

R. Johannesson and E. Paaske, "Further Results on Binary Convolutional Codes with an Optimum Distance Profile," *IEEE Trans. Inform. Theory*, Vol. IT-24, pp. 264-268, March 1978.

L.R. Bahl, and F. Jelinek, "Rate 1/2 Convolutional Codes with Complementary Generators," *IEEE Trans. Inform. Theory*, Vol. IT-17, pp. 718-727, November 1971.

E. Paaske, "Short Binary Convolutional Codes with Maximal Free Distance for Rates 2/3 and 3/4," *IEEE Trans. Inform. Theory*, Vol. IT-20, pp. 683-698, September 1974.

J.B. Cain, G.C. Clark, Jr., and J.M. Geist, "Punctured Convolutional Codes of Rate $(n-1)/n$ and Simplified Maximum Likelihood Decoding," *IEEE Trans. Inform. Theory*, Vol. IT-25, pp. 97-100, January 1979.

J.L. Massey, and D.J. Costello, Jr., "Nonsystematic Convolutional Codes for Sequential Decoding in Space Applications," *IEEE Trans. Comm. Technol.*, Vol. COM-19, pp. 806-813, October 1971.

L.N. Lee, "Short Unity-Memory Byte-Oriented Binary Convolutional Codes having Maximal Free Distance," *IEEE Trans. Inform. Theory*, Vol. IT-22, pp. 349-352, May 1976.

D.G. Daut, J.W. Modestino, and L.D. Wismer, "New Short Constraint Length Convolutional Codes Constructions for Selected Rational Rates," *IEEE Trans. Inform. Theory*, Vol. IT-28, pp. 794-800, September 1982.

A.J. Viterbi, and J.K. Omura, *Principles of Digital Communication and Coding*, McGraw-Hill, 1979.

R. Palazzo, Jr., "Analysis of Periodic Linear and Nonlinear Trellis Codes," Ph.D. Dissertation, University of California, Los Angeles, 1984.

R. Palazzo Jr., "New Short Constraint Length Convolutional Codes Derived From a Network Flow Approach," *Abstracts of Papers*, International Symposium on Information Theory, Brighton England, Julho, 1985.

R. Palazzo Jr., "On the Relationship between the Minimum Distance of Convolutional Codes and the Maximum Flow in Networks," 4º Simpósio Brasileiro de Telecomunicações, RJ, Setembro, 1985.

R. Palazzo Jr., "A New Algorithm for the Combinatorial Optimization Problem of Convolutional Codes," a ser submetido ao *IEEE Transactions on Information Theory*

R. Palazzo Jr., "The Inherent Computational Complexity of Good Convolutional Codes," a ser submetido ao *IEEE Transactions on Information Theory*.

J.A. Heller, "Short Constraint Length Convolutional Codes," *Jet Propulsion Lab., California Inst. Technol., Space Programs Summary* 37-54, Vol. 3, pp. 171-177, Oct./Nov. 1968.

L.R. Ford, and D.R. Fulkerson, *Flow in Networks*, Princeton University Press (Princeton, N.J.), 1962.

G.S. Lauer, "Some Optimal Partial Unit-Memory Codes," *IEEE Trans. Inform. Theory*, Vol. IT-25, pp. 240-243, March 1979.

S. Lin, and D.J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, Prentice Hall, 1983.

R. Palazzo Jr., "A General Relationship Between the Maximum Flow in Networks and the Minimum Distance of Convolutional Codes over GF(q), *3rd. Joint USSR-Swedish International Workshop on Information Theory Convolutional Codes: Multi-User Communication*", Maio 24-30, 1987, Sochi, USSR.

CAPÍTULO 3

CÓDIGOS CONVOLUCIONAIS VARIANTES NO TEMPO

3.1 - INTRODUÇÃO

O que vimos no Capítulo 2 foi basicamente um novo método de descrição e análise de códigos convulucionais específicos e invariantes no tempo.

A especificidade está diretamente relacionada com o número de dígitos que deverão ser acrescentados aos dígitos a serem codificados. No caso em questão o comprimento dos dígitos a serem acrescentados iguala a memória do codificador.

A invariância no tempo implica que as conexões entre os registros e somadores mod q são mantidas ao longo de toda a operação de codificação.

Neste capítulo, estaremos abordando aspectos de determinação de códigos convolucionais específicos, porém, com as conexões variando com o tempo de uma forma periódica.

As consequências dessa alternância serão detalhadas na Seção 3.2 onde a função enumeradora destes códigos será apresentada.

Na Seção 3.3, iremos mostrar como através da estrutura matemática estabelecida podemos determinar códigos ótimos variantes no tempo e compará-los com os códigos ótimos invariantes no tempo.

Na Seção 3.4 apresentaremos as condições necessárias para que as características de proteção desigual aos bits a serem codificados possam ser exploradas com eficiência. Dois critérios serão discutidos e dada a "equivalência" entre os mesmos, um só será utilizado na análise.

3.2 - FUNÇÃO ENUMERADORA DE CÓDIGOS CONVOLUCIONAIS VARIANTES NO TEMPO

A metodologia utilizada na sistematização da enumeração de códigos

convolucionais invariantes no tempo foi a de modelar como um sistema linear dinâmico e discreto.

O mesmo procedimento será aqui utilizado, porém, adaptado à periodocidade das equações de estado e de saída.

Para o estabelecimento das equações de estado e de saída a seguinte propriedade é importante.

Propriedade 1 : Se $\underline{x}_i$ e $\underline{x}_j$ são palavras de um código linear C, então $\underline{x}_i \oplus \underline{x}_j$ também será.

Seja

$$\underline{x}_i = \underline{u}_i \cdot G \qquad e \qquad \underline{x}_j = \underline{u}_j \cdot G$$

então

$$\underline{x}_i \oplus \underline{x}_j = \underline{u}_i \cdot G \oplus \underline{u}_j \cdot G$$

$$= (\underline{u}_i \oplus \underline{u}_j) \cdot G$$

Como $\underline{u}_i$ e $\underline{u}_j$ são dados de entrada n-dimensionais, sua soma mod q também será um dado de entrada n-dimensional, pois os q^n dados de entrada devem coincidir com todos os possíveis vetores n-dimensionais. Assim, se

$$\underline{u}_i \oplus \underline{u}_j = \underline{u}_p$$

segue que

$$\underline{x}_i \oplus \underline{x}_j = \underline{u}_p \cdot G = \underline{x}_p$$

que, portanto, é uma palavra do código.

Agora indo um pouco mais além, seja C um código constituído de M palavras n-dimensionais, isto é, $C = \{\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_M\}$.

Da Propriedade 1, temos

$$\{\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_M\} = \{\underline{x}_1 \oplus \underline{x}_k, \ \underline{x}_2 \oplus \underline{x}_k, \ \ldots, \ \underline{x}_M \oplus \underline{x}_k\}$$

onde $k = 1, 2, \ldots, M$ e $\underline{x}_1 = (0, 0, \ldots, 0) = \underline{0}$

Seja $w(\underline{x}_i)$ o peso de Hamming da palavra $\underline{x}_i$. Seja $w_k = w(\underline{x}_1, \underline{x}_k) = w(\underline{0}, \underline{x}_k)$ para todo k inteiro menor ou igual a M.

Da Propriedade 1, o conjunto $W = \{w_1, w_2, \ldots, w_M\}$ de todas as palavras do código não nula iguala o conjunto $W = \{w(\underline{x}_i, \underline{x}_j) : 1 \leqslant j \leqslant M\}$ com $1 \leqslant i \leqslant M$.

Como podemos associar um caminho na treliça às palavras do código C, então acabamos de mostrar que os códigos convolucionais variantes no tempo são códigos lineares.

Dessa forma, o diagrama de estado pode ser modificado de modo a fornecer uma descrição dos pesos de Hamming de todas as palavras não nulas do código.

De uma forma geral, considere dois sistemas lineares discretos L_1 e L_2. Iremos supor que a evolução temporal de tais sistemas siga o seguinte modo de operação: o sistema L_1 entra em operação no intervalo de tempo $i \leqslant t \leqslant i+1$ e o sistema L_2 no intervalo de tempo $i+1 \leqslant t \leqslant i+2$, isto é, os sistemas L_1 e L_2 operam ciclicamente.

As equações de estado e de saída para este processo podem ser descritas matematicamente através do seguinte argumento: um estado qualquer no sistema L_i, $i = 1, 2$, no instante de tempo $t = u$ pode ser alcançado a partir de qualquer estado no sistema anterior L_j, no instante de tempo $t = u - 1$, pela aplicação da matriz de transição, A_i, correspondente ao sistema L_i, ao estado pertencente a L_j ou a partir de uma condição inicial B_i.

A equação de saída T_i do sistema L_i é obtida pela multiplicação da matriz H_i pela matriz de estado do sistema L_j.

Matematicamente temos

Sistema L_1

$$E_1(i+1) = A_1(u) \cdot E_2(u) + B_1 \tag{3.1}$$

$$T_1(u) = H_1(u) \cdot E_2(u) \tag{3.2}$$

Sistema L_2

$$E_2(u+1) = A_2(u) \cdot E_1(u) + B_2 \tag{3.3}$$

$$T_2(u) = H_2(u) \cdot E_1(u) \tag{3.4}$$

A equação de saída total é dada por

$$T(u) = T_1(u) + T_2(u) \tag{3.5}$$

Resolvendo-se (3.1) e (3.3) e substituindo em (3.2) e (3.4), a equação (3.5) fica completamente determinada. As definições de $E_i(u)$, $A_i(u)$, $H_i(u)$, B_i e $T_i(u)$ seguem exatamente aquelas apresentadas no Capítulo 2 para sistemas invariantes no tempo.

Devemos salientar que a recursividade da solução da equação de estado para sistemas variantes no tempo segue exatamente àquela para sistemas invariantes no tempo com o cuidado especial ao fato da existência da dependência da condição inicial quanto ao sistema L_1 e L_2.

Para o sistema L_1 teremos

$$E_1(u+1) = A_1(u)A_2(u-1)E_1(u-1) + A_1(u) \cdot B_2 + B_1$$

para a primeira transição.

Para o sistema L_2, teremos

$$E_2(u+1) = A_2(u) \cdot A_1(u-1)E_2(u-1) + A_2(u)B_1 + B_2$$

também para a primeira transição.

Com a evolução deste processo no tempo, notamos que a matriz de transição resultante de cada sistema são multiplicadas seguindo uma determinada ordem e que, portanto, uma atenção especial tem que ser tomada uma vez que multiplicação de matrizes não é comutativa.

De modo a termos um tratamento mais formal do problema em consideração, iremos primeiramente estabelecer algumas notações que serão utilizadas no transcorrer do desenvolvimento da função enumeradora.

Seja $\underline{x}_{u,v} = (x_u, x_{u+1}, \ldots, x_v)$ uma sequência de dados de entrada com início no instante de tempo $t = u$ em algum estado S_k pela aplicação de uma operação G_i e terminando no estado S_j no instante de tempo $t = v$ com $v - u >> \nu + 1$ e ν a memória.

Seja G_i com $i = 1, 2$ duas transformações lineares a serem aplicadas de uma forma periódica ao conteúdo dos registros, isto é, iremos supor que no instante $t = 2u$ e $t = 2u + 1$, b bits de informação são armazenados nos registros e que G_1 e G_2 são aplicadas produzindo n símbolos.

Note que $\underline{x}_{u,v}$ gera um caminho na treliça onde cada palavra do código de ramo está sendo gerada pela aplicação de G_1 e/ou G_2.

Da Propriedade 1 e como G_1 e G_2 são transformações lineares, o código de saída terminado no instante $t = v$ forma um grupo, portanto, o código é dito um código de grupo. A consequência deste fato é que "a sequência de dados de

entrada toda nula" pode ser invocada. Então, poderemos representar cada codificador de estado finito por um diagrama de estado particionado juntamente com uma funcional descrevendo a seleção periódica da saída dos codificadores.

Suponhamos que cada codificador de estado finito tenha taxa sem dimensão $r = \frac{1}{n}$. Seja z uma variável independente. Defina α_i como sendo a função distorção, isto é,

$$\alpha_i = \begin{cases} 1 , & \text{se } \underline{x}_{u,v}(k) \neq \hat{\underline{x}}_{u,v}(k) \\ 0 , & \text{se } \underline{x}_{u,v}(k) = \hat{\underline{x}}_{u,v}(k) \end{cases}$$

onde $\underline{x}_{u,v}(k)$ significa o k-ésimo dado de entrada da sequência $\underline{x}_{u,v}$ e $\hat{\underline{x}}_{u,v}(k)$ o k-ésimo dado de entrada estimado da sequência $\hat{\underline{x}}_{u,v}$.

Seja $\underline{\xi}_i$, $\underline{B}_i$, $\underline{H}_i$ e A_i matrizes com dimensões $(M-1) \times 1$, $(M-1) \times 1$, $1 \times (M-1)$ e $(M-1) \times (M-1)$ representando os estados internos, condição inicial, condição de saída e transição entre os estados do codificador de estados finito com $i = 1, 2$ e $M = 2^{b\nu}$.

No início desta seção foi levado em consideração o caso onde a periodicidade era 2. Iremos agora generalizar para uma periodicidade arbitrária W. Novamente, o objetivo é a representação de cada sistema através de suas equações de estado e de saída. Para tal, seja W a periodicidade da combinação dos sistemas lineares discretos L_i com $1 \leqslant i \leqslant W$. Seja $\underline{\xi}_i$ os estados internos no sistema L_i no instante $t = p$. Estes estados podem ser atingidos a partir de qualquer estado interno do sistema $L_{(i-1) \bmod W}$ no instante $t = p - 1$ pela aplicação da matriz de transição, A_i, correspondente ao sistema L_i ou a partir da condição inicial $\underline{B}_i$.

Matematicamente isto pode ser descrito por

$$\underline{\xi}_i = \underline{B}_i + A_i \cdot \underline{\xi}_{(i-1) \bmod W} \quad , \qquad 1 \leqslant i \leqslant W \tag{3.6}$$

A equação de saída T_i do sistema L_i é obtida pela multiplicação da matriz condição de saída $\underline{H}_i$ pela matriz dos estados internos $\xi_{(i-1) \bmod W}$ do sistema $L_{(i-1) \bmod W}$. Isto é

$$T_i = \underline{H}_i \cdot \underline{\xi}_{(i-1) \bmod W} \quad , \qquad 1 \leqslant i \leqslant W \tag{3.7}$$

É de conhecimento geral que a probabilidade de erro de bit é limitada superiormente por

$$P_b \leqslant \frac{1}{2b} \frac{1}{dz} T(D, z, \alpha_1, \alpha_2, \ldots, \alpha_W)\Big|_{z=1} \tag{3.8}$$

Quando $r = \frac{1}{n}$, temos que cada palavra do código de ramo está associada a um bit de informação. Como $\alpha_1, \alpha_2, \ldots, \alpha_W$ são eventos mutuamente exclusivos, isto é, quando um erro de decodificação ocorre no sistema L_i um erro de decodificação não pode ocorrer no sistema L_j, $j \neq i$, uma vez que a decodificação no sistema L_j ainda está para acontecer. Consequentemente, $\alpha_i = 1$, $\alpha_j = 0$ para $j \neq i$.

Se fixarmos o dado de entrada como "0" na sequência toda nula então, $P_b(\alpha_1, 0, \ldots, 0)$ é a função enumeradora de todos os eventos de erro cujo bit do dado de entrada decodificado é "1" no instante $t = 2u$; para todos os caminhos que divergiram do caminho correto antes do instante $t = 2u$ e remergiram posteriormente para o caminho correto.

Representaremos tais funções por

$$P_{b_i} \leq P_b(0, \ldots, \alpha_i, \ldots, 0) \leq \frac{1}{2b} \frac{d}{dz} T(D, z, \alpha_1, \ldots, \alpha_W)\Big|_{z=1,\alpha_i=1,\alpha_j=0} \qquad (3.9)$$

para $j \neq i$.

Como esses eventos são disjuntos, a probabilidade de erro de bit total é, pela união de eventos, dada por

$$P_b = \sum_{i=1}^{W} P_{b_i}$$

Explicitamente, a solução de (3.6), (3.7) e (3.8) é a seguir apresentada para um W qualquer.

$$\underline{\xi}_W = \left(I - \prod_{i=0}^{W-1} A_{W-i}\right)^{-1} \cdot \left(\sum_{\gamma=1}^{W-1} \left[\prod_{m=0}^{\gamma-1} A_{W-m}\right] B_{W-\gamma} + B_W\right) \qquad (3.10)$$

$$\underline{\xi}_k = \left(\prod_{i=0}^{k-1} A_{k-i}\right) \cdot \left(I - \prod_{i=0}^{W-1} A_{W-i}\right)^{-1} \cdot \left(\sum_{\gamma=1}^{W-1} \left[\prod_{m=0}^{\gamma-1} A_{W-m}\right] B_{W-\gamma} + B_W\right) +$$

$$+ \left(\sum_{n=1}^{k-1} \prod_{i=0}^{k-1-n} A_{k-1}\right) B_n + B_k \quad , \quad 1 \leq l \leq W-1 \qquad (3.11)$$

$$T_n(D, z, \alpha_1, \ldots, \alpha_W) = \frac{d}{dz} H_{n+1} \cdot \underline{\xi}_n + H_{n+1} \cdot \frac{d}{dz} \underline{\xi}_n \qquad (3.12)$$

e

$$P_b \leq \left(\frac{1}{2bW}\right) \left[\sum_{n=1}^{W-1} T_n(D, z, \alpha_1, \ldots, \alpha_W) + \frac{d}{dz} H_1 \cdot \underline{\xi}_W + H_1 \cdot \frac{d}{dz} \underline{\xi}_W\right] \qquad (3.13)$$

3.3 - CÓDIGOS CONVOLUCIONAIS VARIANTES NO TEMPO

Desde a conjectura apresentada por Costello, isto é, que os códigos convolucionais não sistemáticos e variantes no tempo possuem distâncias livres maiores do que as distâncias livres dos códigos convolucionais invariantes no tempo, tem conduzido um número de pesquisadores à tarefa de determinação de tais códigos com a finalidade de confirmar tal conjectura.

Mooser determinou que algumas combinações periódicas de códigos convolucionais invariantes no tempo atingem os mesmos valores das distâncias livres dos códigos convolucionais ótimos invariantes no tempo, porém, com um número médio de caminhos com peso d_∞ por instantes de tempo, N_∞, e um número médio de bits de informação em erro por instante de tempo ao longo dos caminhos com peso d_∞, I_∞, menores do que os invariantes no tempo. Entretanto, essas combinações periódicas são somente para o caso em que a memória $\nu = 4$, taxa $r = 1/2$ e com período até 5.

Nesta seção, iremos apresentar uma combinação periódica com memória $\nu = 1$ e taxa $r = 2/3$ com distância livre maior do que a distância livre do melhor código convolucional invariante no tempo, bem como combinações periódicas com qualquer período T de códigos convolucionais invariantes no tempo com $\nu = 2$ e taxas $r = 1/(3n + 1)$ para $n \geq 1$; algumas combinações periódicas com $\nu = 4$, taxa $r = 1/2$ e período $T = 2$ e 3 tal que a distância livre é a mesma mas com N_∞ e I_∞ menores do que N_∞ e I_∞ dos melhores códigos invariantes no tempo.

3.3.1 - Preliminares

De modo a utilizar uma notação característica de um grupo de pesqui-

sadores em códigos convolucionais sem que novos símbolos sejam introduzidos iremos a seguir estabelecer explicitamente tais parâmetros.

Sejam i_u e t_u a representação das b-uplas e n-uplas dos dígitos de entrada e saída, respectivamente, no instante de tempo u para o codificador convolucional binário com $u = 0, 1, 2, \ldots$.

Seja ν a memória de um codificador variante no tempo descrito pelas matrizes $G_j(u)$, com dimensão $b \times n$, $0 \leqslant j \leqslant \nu$ e $u = 0, 1, 2, \ldots$, tal que

$$t_u = \sum_{j=0}^{\nu} i_{u-j} \, G_j(u) \, , \qquad u = 0, 1, 2, \ldots \tag{3.14}$$

onde $i_u = \underline{0}$ para $u < 0$.

Se $G_j(u) = G_j$ para todo j e u, então o codificador é dito ser um codificador convolucional invariante no tempo.

Se $G_j(u) = G_j(u + T)$ para algum T positivo, para todo j e u, então o codificador é dito ser um codificador convolucional periodicamente variante no tempo com período T.

Sejam $i_{[u,v)}$ e $t_{[u,v)}$ a representação das sequências de entrada e saída, respectivamente, sobre um intervalo de tempo que inicia em u e termina em $v - 1$, isto é, $u, u + 1, u + 2, \ldots, v - 2, v - 1$.

Seja d_∞ a distância livre de um código convolucional, isto é, a menor distância de Hamming entre todos os pares de sequências de saída, $t_{[0,\infty)}$ resultantes das distintas sequências de entrada $i_{[0,\infty)}$. Equivalentemente, d_∞ é o menor $w(t_{[0,\infty)})$ que possa ser obtido quando $i_{[0,\infty)} \neq 0$, com $w(\cdot)$ sendo o peso de Hamming. Para os códigos convolucionais periodicamente variantes no tempo, d_∞ é equivalente ao menor $w(t_{[0,\infty)})$ que possa ser obtido quando $i_{[0,T)} \neq 0$.

Um fato que pode ocorrer é que tanto os códigos convolucionais invariantes no tempo quanto os periodicamente variantes no tempo apresentem a mesma distância livre. Deste modo, de forma a resolver esse impasse, mesma d_∞, algumas alternativas devem ser definidas.

Defina N_∞ como sendo o número médio dos caminhos com distância d_∞ por instantes de tempo e I_∞ como o número médio de bits de informação errôneos por instante de tempo ao longo dos caminhos com distância d_∞.

Com estas definições as alternativas para solucionar o impasse mencionado acima ficam estabelecidas, a saber, a primeira alternativa está relacionada com N_∞ e a segunda com I_∞.

Matematicamente, N_∞ e I_∞ são definidos por

$$N_\infty = \frac{1}{T}\left\{ i_{[0,\infty)} : i_{[0,T)} \neq 0 \quad \text{e} \quad w(t_{[0,\infty)}) = d_\infty \right\} \tag{3.15}$$

e

$$I_\infty = \frac{1}{T} \sum_{i_{[0,\infty)} \in S} w(i_{[0,\infty)}) \tag{3.16}$$

respectivamente, onde S é o conjunto de sequências de entrada à direita de (3.15).

Poderíamos ter chegado à (3.15) e (3.16) através da técnica de função de transferência dinâmica que por sua vez é uma generalização da técnica de função de transferência para códigos convolucionais invariantes no tempo.

3.3.2 - Os Novos Códigos

Da sub seção 3.3.1 temos que o critério de otimalidade a ser adotado

quando da comparação dos códigos convolucionais periodicamente variantes no tempo com período T com os códigos convolucionais invariantes no tempo será baseado nos parâmetros d_∞, N_∞ e I_∞.

Definiremos a superioridade de um código convolucional periodicamente variante no tempo com período T com relação à um código convolucional invariante no tempo se aquele:

1) possuir maior d_∞; ou

2) possuir menores valores para N_∞ e I_∞ quando do impasse da distância d_∞.

Baseado neste critério de otimalidade e sabendo que a estrutura de treliça gerada por (3.14) é cíclica, apresentaremos a seguir um código convolucional periodicamente variante no tempo com período T = 2 apresentando uma distância livre maior que a distância livre do código convolucional invariante no tempo. Dessa forma, satisfazendo a condição 1) do critério de otimalidade. Esta combinação periódica consiste de dois códigos invariantes no tempo cada com memória $\nu = 1$, taxa $r = 2/3$ e período $T = 2$ com matrizes geradoras $G^1 = \{G_0^1, G_1^1\}$ e $G^2 = \{G_0^2, G_1^2\}$ respectivamente, dadas por

$$G_0^1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad G_1^1 = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

e

$$G_0^2 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \qquad G_1^2 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Tabela 1

Memória $\nu = 2$

T	r	d_∞	N_∞	I_∞	representação octal				
1	1/4	10	1	1	$5^2\ 7^2$				
2	1/4	10	1/2	1/2	$5^2\ 7^2$	$5\ 7^3$			
3	1/4	10	2/3	2/3	$5^2\ 7^2$	$5^2\ 7^3$			
4	1/4	10	3/4	3/4	$5^2\ 7^2$	$5^2\ 7^2$	$5^2\ 7^2$	$5\ 7^3$	
5	1/4	10	4/5	4/5	$5^2\ 7^2$	$5^2\ 7^2$	$5^2\ 7^2$	$5^2\ 7^2$	$5\ 7^3$
1	1/7	18	1	1	$5^3\ 7^4$				
2	1/7	18	1/2	1/2	$5^3\ 7^4$	$5^2\ 7^5$			
3	1/7	18	2/3	2/3	$5^3\ 7^4$	$5^3\ 7^4$	$5^2\ 7^5$		
4	1/7	18	3/4	3/4	$5^3\ 7^4$	$5^3\ 7^4$	$5^3\ 7^4$	$5^2\ 7^5$	
5	1/7	18	4/5	4/5	$5^3\ 7^4$	$5^3\ 7^4$	$5^3\ 7^4$	$5^3\ 7^4$	
1	1/10	26	1	1	$5^4\ 7^6$				
2	1/10	26	1/2	1/2	$5^4\ 7^6$	$5^3\ 7^7$			
3	1/10	26	2/3	2/3	$5^4\ 7^6$	$5^4\ 7^6$	$5^3\ 7^7$		
4	1/10	26	3/4	3/4	$5^4\ 7^6$	$5^4\ 7^6$	$5^4\ 7^6$	$5^3\ 7^7$	
5	1/10	26	4/5	4/5	$5^4\ 7^6$	$5^4\ 7^6$	$5^4\ 7^6$	$5^4\ 7^6$	$5^3\ 7^7$

Tabela 2

Memória $\nu = 4$

T	r	d_∞	N_∞	I_∞	Representação Octal					
1	1/2	7	2	4	23	35				
2	1/2	7	3/2	3	23	35	25	37		
3	1/2	7	4/3	8/3	23	35	25	37	27	35

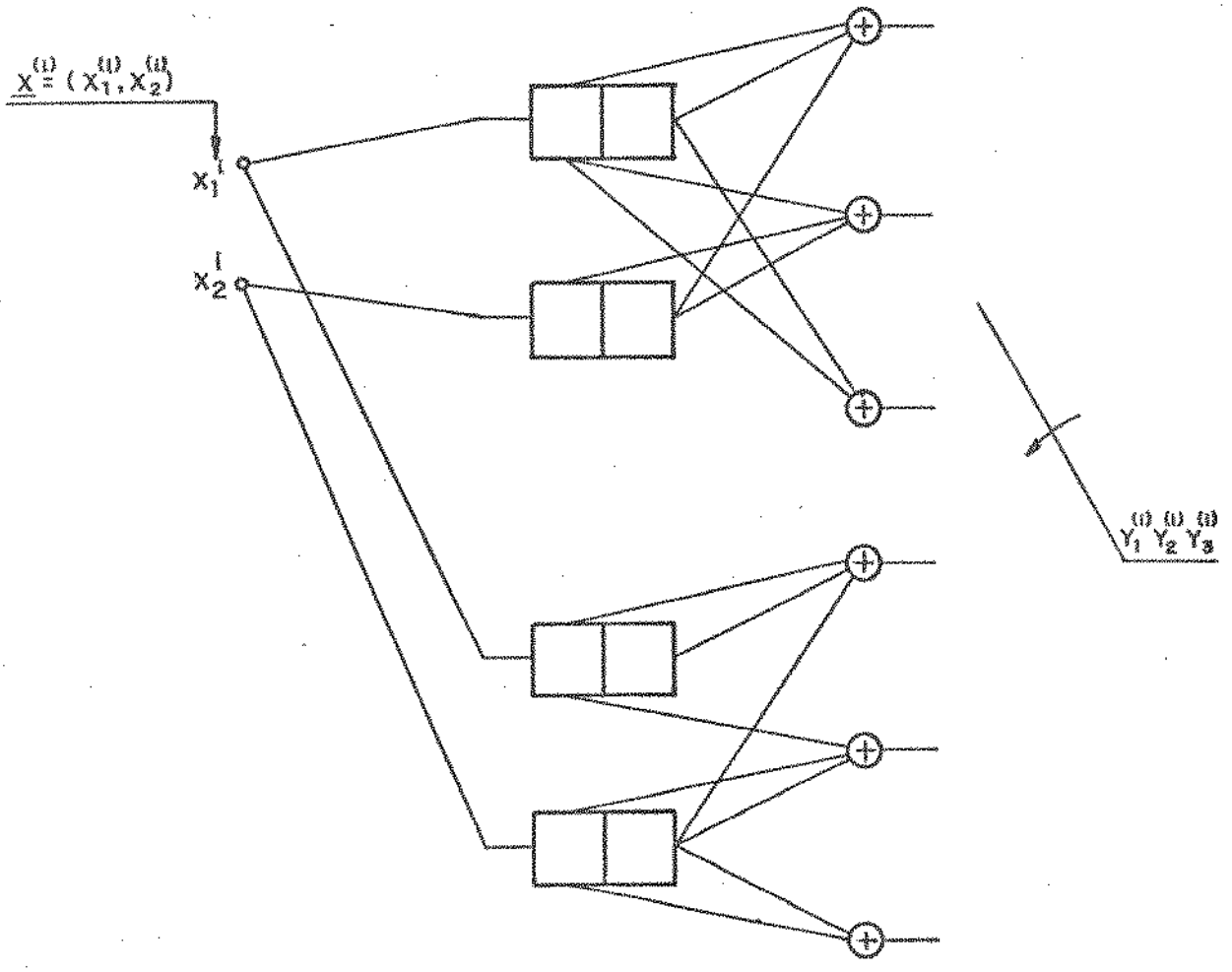


Figura 3.3.2.1 - Codificador periodicamente variante no tempo com $T = 2$.

resultando em um código convolucional periodicamente variante no tempo e não catastrófico. (ver Fig. 3.3.2.1)

Qualquer um desses códigos convolucionais invariantes no tempo é considerado *ótimo* com respeito ao critério de máxima distância livre. Os parâmetros d_∞, N_∞ e I_∞ assumem os seguintes valores $d_\infty = 3$, $N_\infty = 1$ e $I_\infty = 1$. Entretanto, o código convolucional periodicamente variante no tempo com período $T = 2$ apresenta $d_\infty = 4$, $N_\infty = 21/2$ e $I_\infty = 57/2$. Este é o único código periodicamente variante no tempo, conhecido, apresentando distância livre maior do que a distância livre do melhor código invariante no tempo.

Uma das maneiras de se chegar ao valor de d_∞ seria: 1) utilização dos algoritmos padrões empregados na determinação dos códigos invariantes no tempo; 2) o procedimento sistemático da técnica de função de transferência dinâmica; e 3) utilizar do fato de que todo código periodicamente variante no tempo com período T possue um código invariante no tempo equivalente.

A seguir apresentamos nas Tabelas 1 e 2 os códigos ótimos periodicamente variantes no tempo com período T com os respectivos valores de N_∞ e I_∞.

Em geral, temos que para qualquer $n > 1$, n um inteiro, os códigos periodicamente variantes no tempo com período $T > 2$ com taxa $r = 1/(3n+1)$ possuem distâncias livres $d_\infty = 8n+2$, $N_\infty = I_\infty = (T-1)/T$, e funções geradoras consistindo de $(T-1)5^{n+1}\,7^{2n}$ e um $5^{n}\,7^{2n+1}$ códigos invariantes no tempo, onde $5^2\,7^2 = 5577$. Para $T = 1$, $d_\infty = 8n+2$, $N_\infty = I_\infty = 1$ e as funções geradoras são dadas por $5^{n+1}\,7^{2n}$.

3.4 - CÓDIGOS CONVOLUCIONAIS COM PROTEÇÃO DE ERRO DESIGUAL

Nesta seção, novas classes de códigos lineares com proteção desigual

derivados de códigos convolucionais serão apresentados.

Estas classes consistem de códigos convolucionais específicos invariantes no tempo com taxas $r = b/n$, $b \geq 2$ e códigos convolucionais específicos periodicamente variantes no tempo com taxas $r = b/n$, $b \geq 1$.

Empregaremos a função distorção média na seleção e avaliação dos códigos lineares com proteção desigual ótimos.

Esta classe de códigos sofre dos mesmos defeitos apresentados pelos códigos convolucionais tradicionais, isto é, a falta de uma estrutura algébrica faz com que procedimentos sistemáticos de geração de tais códigos fiquem inviabilizados. Consequentemente, tais códigos só poderão ser determinados através de buscas exaustivas.

3.4.1 - Introdução

Desde a introdução de códigos lineares com proteção desigual por Masnick Wolf, muitos pesquisadores tem estendido e derivado resultados novos para classes de códigos lineares com proteção desigual para os casos de proteção desigual em posições únicas nas palavras do código e posições únicas dos dígitos de informação.

Estas classes de códigos consistem de códigos com proteção desigual cíclicos não sistemáticos, códigos derivados de diferença de conjuntos, "design" iterativos e concatenados de proteção desigual, classes de códigos cíclicos e códigos lineares de proteção desigual derivados de códigos com comprimento menores.

O objetivo deste seção é o de apresentar classes de códigos convolu-

lucionais que satisfaçam a propriedade básica de códigos lineares com proteção desigual, isto é, fornece proteção desigual para cada dígito de informação. Estas classes de novos códigos consistem essencialmente de códigos convolucionais específicos não sistemáticos e invariantes no tempo bem como periodicamente variantes no tempo.

Vários pesquisadores estabeleceram, para códigos lineares de bloco, a existência de codificação ótima e procedimentos sistemáticos para a construção de códigos de proteção desigual para posições únicas nas palavras do código bem como para posições únicas dos dígitos de informação, através do uso da matriz geradora ou do uso da matriz de paridade juntamente com o processo de decodificação, que em geral é o processo de decodificação por lógica majoritária.

Em contraste com as classes de códigos lineares com proteção desigual previamente estabelecidas, as novas classes não apresentam uma estrutura algébrica, portanto, bons códigos só poderão ser determinados através de busca exaustiva. Entretanto, o processo de decodificação continua a empregar o algoritmo de Viterbi para restrições de memória razoáveis e qualquer dos algoritmos para decodificação sequencial para restrições de memória grandes.

3.4.2 - Critérios de Seleção e Avaliação

O critério a ser utilizado na seleção e avaliação dos códigos convolucionais com proteção desigual nas posições únicas dos dígitos de informação é a média da função distorção, distorção média, a qual é equivalente ao conceito do vetor separação dos códigos lineares de bloco (n, k), com k o número de dígitos de informação e n o comprimento da palavra-código, introduzido por

Dunning e Robbins.

Iremos representar um código convolucional com memória ν e taxa $r = b/n$ por (b, n, ν).

Definição 2 : Para um código convolucional específico invariante no tempo representado por (b, n, ν) com $b \geq 2$ e códigos convolucionais específicos periodicamente variante no tempo sobre GF(2), o vetor de distorção média $d(C)$, $d(C) = \{d(1), d(2), \ldots, d(b)\}$ será definido por

$$d(i) = \min \text{ grau} \left\{ \frac{d}{dz} T(\alpha_1, \ldots, \alpha_b, z, D) \right\}, \qquad 1 \leq i \leq b$$

onde $z = 1$, $\alpha_i = 1$ e $\alpha_j = 0$, $j \neq i$, $T(\cdot)$ é a função de transferência para os códigos invariantes no tempo ou a função de transferência dinâmica para os códigos periodicamente variantes no tempo, z é um erro de decodificação, D é a função Bhattacharyya e os α_i's são funções distorções tal que $\alpha_i = 0$ se os i-ésimos dígitos de informação transmitido e decodificado são iguais e $\alpha_i = 1$ se forem diferentes.

Embora nesta definição tenhamos utilizado a técnica de função de transferência, a qual se ajusta perfeitamente ao problema, a matriz geradora semi-infinita G poderia também ser utilizada.

Sob este prisma teríamos G representada por

$$G = \begin{bmatrix} G_o & G_1 & \cdots & G_\nu & & \\ & G_o & G_1 & \cdots & G_\nu & \\ & & \ddots & & \ddots & \\ & & & G_o & G_1 & \cdots & G_\nu \end{bmatrix}$$

onde as áreas em branco são iguais a zero.

Desta forma teríamos

Definição 3 : Para um código convolucional específico invariante no tempo representado por (n, n, ν) com b ≥ 2 e código convolucional periodicamente variante no tempo sobre GF(2), o vetor separação d(C), d(C) = {d(1), ..., d(b)} será definido por

$d(j) = \min \{ w(\underline{u} \cdot G)$: para cada posição j fixada somente em $u_o \neq 0$, $1 \leq j \leq b$, e as posições restantes assumindo qualquer um dos $2^b - 1$ valores com $u_i \in GF(2)^b \}$.

onde $\underline{u} = (u_o, u_1, u_2, \ldots)$ é a sequência dos dígitos de informação com $u_i = (u_{i,1}, u_{i,2}, \ldots, u_{i,b})$ e $w(\cdot)$ é o peso de Hamming.

As definições 2 e 3 podem ser facilmente estendidas para o caso em que o corpo de Galois possue q elementos GF(q), com q um número primo ou uma potência de um número primo.

Embora ambas definições conduzam a critérios equivalentes, a defini

ção 2 vai além da definição 3 em termos de a mesma fornecer a função enumeradora dos pesos dos bits, a qual é o objetivo principal no processo de seleção e avaliação dos códigos.

À partir das definições 2 ou 3 pode-se provar a seguinte capacidade de correção de um código convolucional com proteção desigual quando empregado num canal com entradas q-árias e saída simétricas.

Teorema 4 : Um código convolucional específico representado por (b, n, ν) com $b \geq 2$ e um código convolucional específico periodicamente variante no tempo sobre GF(2) com matriz geradora G e decodificação de Viterbi ou Sequencial, garante que o i-ésimo dígito de informação será decodificado corretamente sempre que o padrão, de erro apresentar uma distância de Hamming menor ou igual a $\lfloor (d(i) - 1)/2 \rfloor$ onde $\lfloor \gamma \rfloor$ representa o maior inteiro menor ou igual a γ.

É evidente da definição 2 ou 3 que a distância mínima do código igual a

$$d_{min} = \min \{ d(1), d(2), \ldots, d(b) \}$$

Se o código convolucional com matriz geradora G resulta em um vetor separação d(C) tal que suas componentes não são mutuamente iguais, então denominaremos este código um *código convolucional com proteção desigual com relação à posição única dos dígitos de informação*.

Definiremos $\tilde{d}(C) = \{\tilde{d}(1), \tilde{d}(2), \ldots, \tilde{d}(b)\}$ como o conjunto ordenado do vetor separação d(C) de um código convolucional (b, n, ν).

Assim as condições de otimalidade, equivalência e não catastrófico desta classe de códigos lineares de proteção desigual às posições únicas dos dígitos de informação podem ser estabelecidas como segue:

Definição 5 : Um código convolucional com proteção desigual às posições únicas dos dígitos de informação é *ótimo* se e somente se não existir nenhum outro conjunto ordenado $\tilde{d}(C1)$ que seja lexicograficamente melhor do que $\tilde{d}(C)$.

Definição 6 : Sejam C1 e C2 dois códigos convolucionais com proteção desigual dos dígitos de informação. Esses códigos são ditos *equivalentes* se e somente se o conjunto ordenado dos vetores separação de cada código são iguais, isto é, $\tilde{d}(C1) = \tilde{d}(C2)$.

Da equação (3.10) temos que se

$$\prod_{i=0}^{W-1} A_{W-i} = I$$

onde I é a matriz identidade, então a combinação periódica dos códigos convolucionais específicos e invariantes no tempo é catastrófica.

Definição 7 : Seja $Q = \prod_{i=0}^{W-1} A_{W-i}\Big|_{D=0}$ a *matriz de transição avaltada.*

Seja $\bar{Q} = Q^{\delta}$, onde $1 \leq \delta \leq \nu$ e ν é a memória do código. Então para todo δ, se $\bar{Q}_{ii} \neq 1$, então a combinação não é catastrófica. Por outro lado, se para pelo menos um valor de δ, $\bar{Q}_{ii} = 1$, então a combinação é catastrófica.

3.4.3 - Classes de Códigos Convolucionais com Proteção Desigual

Iniciaremos a determinação das classes de códigos convolucionais com proteção desigual aos dígitos de informação pela restrição das classes dos códigos que não poderão ser usados na proteção desigual. Para tal, temos

Lema 8 : Um código convolucional específico invariante no tempo representado por $(1, n, \nu)$ não é capaz de fornecer proteção desigual.

Demonstração : Considere um código convolucional específico invariante no tempo representado por $(1, n, \nu)$ juntamente com sua treliça. Como este código é linear, sem perda de generalidades, assuma que a sequência toda nula é transmitida. Um erro de decodificação ocorrerá no instante $t = k$ se existir um caminho que divirja do caminho correto (sequência toda nula) no instante $t = k$ ou anterior a ele e convirja para o caminho correto posteriormente.
Seja $\underline{d} = \{d_1, d_2, \ldots\}$ o conjunto ordenado dos pesos de Hamming de todos os caminhos satisfazendo a condição de decodificação errônea acima mencionada.
Agora, assuma que uma decodificação errônea tenha ocorrido no instante $t = k'$ com $k' \neq k$. Seja $\underline{d}' = \{d'_1, d'_2, \ldots\}$ o conjunto ordenado dos pesos de Hamming associados à esta condição de erro.

Como o processo evolutivo é Markoviano e estacionário, as estatísticas associadas a um evento de erro no instante $t = k$ e $t = k'$ são as mesmas. Portanto, os conjuntos $\underline{d}$ e $\underline{d}'$ são iguais e qualquer dígito de informação tem a mesma proteção, ou equivalentemente, possuem a mesma distância mínima.

C.Q.D.

Por outro lado, códigos convolucionais específicos invariantes no tempo com a representação de ν memórias para cada uma das entradas b-paralelas com taxa $r = b/n$, $b \geq 2$, fornecem uma proteção desigual natural para cada um dos b dígitos de informação. Isto acontece principalmente pela *não simetria* das conexões entre os registros e os somadores mod 2.

Na classe dos códigos convolucionais específicos periodicamente variantes no tempo que apresentem o vetor separação com componentes não mutuamente iguais, uma gama de possibilidades de tais códigos existem. Por exemplo podemos determinar bons códigos convolucionais com proteção desigual aos dígitos de informação representados por (b, n, ν) impondo as seguintes condições:

a) a combinação tem a mesma taxa e restrição de memória;

b) a combinação é constituída de códigos com diferentes taxas, porém, com o mesmo máximo fluxo;

c) a combinação é constituída de códigos com diferentes máximos fluxos, porém, com o mesmo n.

Em geral, c) apresenta códigos com o mesmo conjunto ordenado, $\tilde{d}(C)$, do vetor separação que o código ótimo invariante no tempo, porém, com um limi-

tante superior na probabilidade de erro de bit ligeiramente menor que aquele do código ótimo invariante no tempo. Assim, sob o critério adotado esses códigos são equivalentes. Consideraremos somente os códigos sob a condição a).

Conjectura-se que, em geral, os códigos convolucionais periodicamente variantes no tempo não são melhores do que os invariantes no tempo sob o critério da distância mínima, a menos do caso apresentado na sub seção 3.3.2. Entretanto, alguns dos códigos convolucionais periodicamente variantes no tempo ótimos apresentam um limitante superior na probabilidade de erro de bit mais acurado do que o limitante para os invariantes no tempo dito ótimos. Este limitante mais acurado decorre do fato de que alguns dos dígitos de informação apresentam distâncias mínimas maiores implicando, de uma maneira global, em um número de caminhos críticos com um número menor de bits de informação errôneos. Portanto, os códigos satisfazendo esta condição podem ser considerados como candidatos para a classe dos códigos com proteção desigual aos dígitos de informação denominados bons.

3.4.4 - Resultados

Através do critério do vetor separação juntamente com um procedimento de busca, apresentamos nas Tabelas de 1 a 5 alguns dos códigos com proteção desigual denominados bons com as respectivas taxas (R), submatrizes dos códigos convolucionais na representação octal (C), conjunto ordenado do vetor separação ($\bar{d}(C)$).

Tabela 3

Memória $\nu = 1$

taxa	G				$\tilde{d}(G)$
2/3	36		25		(3, 4)
2/5	31	26	25	33	(6, 7)
2/8	370	037	174	237	(10, 11)
2/11	3163	1755	1077	3760	(14, 15)
2/14	37700	03477	37760	01777	(18, 19)
2/17	003777	377700	201777	177760	(22, 23)
2/20	3777600	0017777	0777740	3007777	(26, 27)

e em geral para taxas $r = 2/(3 \cdot n + 5)$; $\tilde{d}(1) = 4n + 6$ and $\tilde{d}(2) = 4n + 7$ para $n = 0, 1, 2, 3, \ldots$

Tabela 4

Memória $\nu = 1$

taxa	G								$\tilde{d}(G)$
3/4	03	05	16		11	12	14		(4, 4, 5)
3/6	70	36	13		34	07	62		(6, 6, 7)
3/8	360	074	227		146	073	303		(8, 9, 9)
3/11	3434	0770	2133		2525	2752	1361		(12, 12, 13)
4/5	16	22	02	13	03	06	14	30	(3, 3, 4, 5)
4/7	170	036	115	055	140	070	056	041	(6, 6, 6, 7)
4/9	740	146	264	453	170	147	716	444	(8, 8, 9, 9)

Tabela 5

Memória $\nu = 3$

Período 2

taxa	$\tilde{d}(C)$	C
1/8	(25, 26)	CC1 - 11 11 13 13 17 17 17 17
		CC2 - 13 13 13 15 15 15 17 17
1/11	(35, 36)	CC1 - 13 13 13 13 15 15
		15 17 17 17 17
		CC2 - 11 13 13 13 13 15
		15 15 17 17 17

Período 3

1/5	(15, 16, 17)	CC1 - 13 13 15 17 17
		CC2 - 11 13 15 17 17
		CC3 - 13 13 15 15 17

Referências

R. Palazzo Jr., "Analysis of Periodic Linear and Nonlinear Trellis Codes", Ph. D. dissertação, UCLA, 1983.

R. Palazzo Jr., "Dynamic Transfer Function Technique Applied to Periodically Time Varying Convolutional Codes", *IEEE International Symposium on Information Theory*, St. Jovite, Quebec, Canadá, 1983.

R. Palazzo Jr., "Bit Error Probabilities of Binary Linear Trellis Codes", *1º Simpósio Brasileiro de Telecomunicações*, Rio de Janeiro, 1983.

R. Palazzo Jr., "Linear Unequal Error Protection Convolutional Codes", *IEEE International Symposium on Information Theory*, Brighton, England, 1985.

R. Palazzo Jr., "On the Linear Unequal Error Protection Convolutional Codes", *IEEE Global Telecommunications 86*, Houston, Texas, USA, 1986.

R. Palazzo Jr., "Time Varying Convolutional Encoders with Larger Free Distance than the Best Time Invariant Encoders", *IEEE Transactions on Information Theory*, aceito para publicação.

R. Palazzo Jr., "New Classes of Linear Unequal Error Protection Codes", submetido ao *IEEE Transactions on Communications*.

B. Masnick e J.K. Wolf, "On Linear Unequal Error Protection Codes", *IEEE Trans. Inform. Theory*, Vol. IT-13, pp. 600-607, Outubro 1967.

L.A. Dunning e W.E. Robbins, "Optimal Encodings of Linear Block Codes for Unequal Error Protection", *Inform. Control*, Vol. 37, pp. 150-177, 1978.

M. Mooser, "Some Periodic Convolutional Codes Better than any Fixed Code", *IEEE Transac. Inform. Theory*, Vol. IT-29, pp. 750-751, Setembro 1983.

D.J. Costello Jr., "Free Distance Bounds for Convolutional Codes", *IEEE Trans. Inform. Theory*, Vol. IT-20, pp. 356-365, Maio 1974.

CAPÍTULO 4

CÓDIGOS DE MEMÓRIA UNITÁRIA: COMPLEXIDADE DE DETERMINAÇÃO E APLICAÇÕES EM SISTEMAS CRIPTOGRÁFICOS

4.1 - INTRODUÇÃO

O problema relativo à determinação de códigos convolucionais invariantes no tempo tem sido alvo de extensivo estudos nos últimos anos dado as suas características de relativa complexidade de implementação do decodificador e um bom índice de performance quando comparadas com as características de códigos de bloco sob as mesmas restrições, ou seja, comprimento das palavras do código e taxa.

Desses estudos realizados teve como resultado proposições de um número expressivo de algoritmos. Entretanto, um método geral para a solução do problema ainda está para ser proposto.

De modo a demonstrar a dificuldade inerente deste problema, na Seção 4.2 iremos mostrar, a partir de um ponto de vista combinatorial, a redução do problema de determinação de bons códigos de memória-unitária e consequentemente que bons códigos convolucionais específicos e não sistemáticos sobre a álgebra de Galois com q elementos, GF(q), onde q é um número primo ou uma potência de um número primo, ao de resolver o problema da mochila (knapsack).

É de conhecimento geral que este problema pertence à classe de problemas denominados Polinomiais Não Determinísticos (NP) Completo, e que é presumivelmente difícil, pelo menos no pior caso. Consequentemente, estaremos justificando a dificuldade de solucionar o problema de determinação dos bons códigos convolucionais.

Como consequência desse resultado, na Seção 4.3 iremos apresentar proposições de novos esquemas criptográficos utilizando os códigos de treliça lineares e não lineares como os elementos de cifragem dos textos a serem criptografados. Na Seção 4.4, iremos abordar as propriedades das funções armadilhas.

Finalmente, na Seção 4.5 a análise da complexidade computacional desses criptosistemas será apresentada.

4.2 - CÓDIGOS DE MEMÓRIA-UNITÁRIA

Antes de entrarmos em consideração sobre a redução do problema de determinação dos códigos ao problema da mochila, iremos estabelecer alguns conceitos pertinentes aos mesmos.

Seja $\underline{x}_t$ um vetor de dados de entrada b-dimensional. Seja $\underline{y}_t$ um vetor de dados codificados n-dimensional sobre o corpo GF(q) definidos por

$$\underline{x}_t = (x_{t1}, x_{t2}, x_{t3}, \cdots, x_{tb})$$

e

$$\underline{y}_t = (y_{t1}, y_{t2}, y_{t3}, \cdots, y_{tn})$$

respectivamente.

Sejam $G_o(t)$ e $G_1(t)$ matrizes b x n variantes no tempo com elementos em GF(q). Um código de memória-unitária (b,n) é definido pela regra de codificação da seguinte forma

$$\underline{y}_t = \underline{x}_t \cdot G_o(t) + \underline{x}_{t-1} \cdot G_1(t), \qquad t > 0 \text{ e } \underline{x}_{-1} = \underline{0}$$

onde $\underline{0}$ é definido como a matriz linha com todos os seus elementos iguais a zero e a operação de soma sendo realizada em GF(q).

Se $G_j(t) = G_j$ para todo t e j, então o código é dito ser invariante

no tempo. Se $G_j(t) = G_j(t+T)$ para todo j e algum T positivo, então o código é dito ser periodicamente variante no tempo. De modo a mostrar a redução é suficiente que o código seja invariante no tempo. Assim, nesta seção iremos adotar esta hipótese.

Iremos denotar o codificador de um código de memória-unitária com taxa de transmissão $r = b/n$ e memória ν pelo paralelismo das b entradas com $(\nu+1)$ estágios de registros. Seja d_∞ a distância livre do código, isto é, o menor peso de Hamming entre pares de sequências de saída resultante de sequências de dados de entrada distintos, ou seja, distância mínima sem restrição.

Como visto no Capítulo 2, em geral, os códigos convolucionais ótimos específicos e não sistemáticos e os códigos ótimos de memória-unitária satisfazem as propriedades de máximo fluxo e conservação de fluxo quando caracterizados como um problema de otimização combinatorial.

Suscintamente, gostaríamos de enfatizar que por *código ótimo* queremos dizer que a função enumeradora dos pesos tem o menor número de palavras com d_∞. Entretanto, se pelo menos dois códigos apresentam o mesmo número de palavras com o valor d_∞, então aquele que apresentar o menor número de palavras com distância $d_\infty + 1$ será considerado ótimo. Se por outro lado, persistir o fato de que ambos os códigos tenham o mesmo número de palavras com distância $d_\infty + 1$, a política adotada será a de continuar o processo de comparação até que um deles apresente um número de palavras menor. Por *máximo fluxo*, Φ, queremos dizer que a soma dos pesos de Hamming dos ramos que deixam e convergem para cada estado na representação no diagrama de estado satisfaz

$$\Phi = n \cdot (q-1) \cdot q^{b-1} \tag{4.1}$$

Note que (4.1) representa o peso total de um código de bloco sobre

GF(q) com comprimento n e q^b palavras. Por *conservação de fluxo* queremos dizer que o fluxo Φ entrando e/ou saindo de cada estado, com exceção do estado zero, na representação de diagrama de estado, são iguais. Consequentemente, as propriedades de máximo fluxo e conservação de fluxo implicam que nenhuma coluna de G_o e G_1 possa ser nula.

Após essa breve recordação do que foi estabelecido no Capítulo 2, estamos aptos a mostrar a equivalência dos problemas mencionados no início desta Seção.

Como os códigos de memória-unitária (bν, nν) são equivalentes aos códigos convolucionais (b,n,ν) e aqueles apresentando pelo menos a mesma distância livre, temos que os códigos de memória-unitária formam uma classe especial de códigos.

Considere que as conexões entre os registros e os somadores mod q sejam arbitrárias para uma dada taxa de transmissão r = b/n. Isto possibilita a

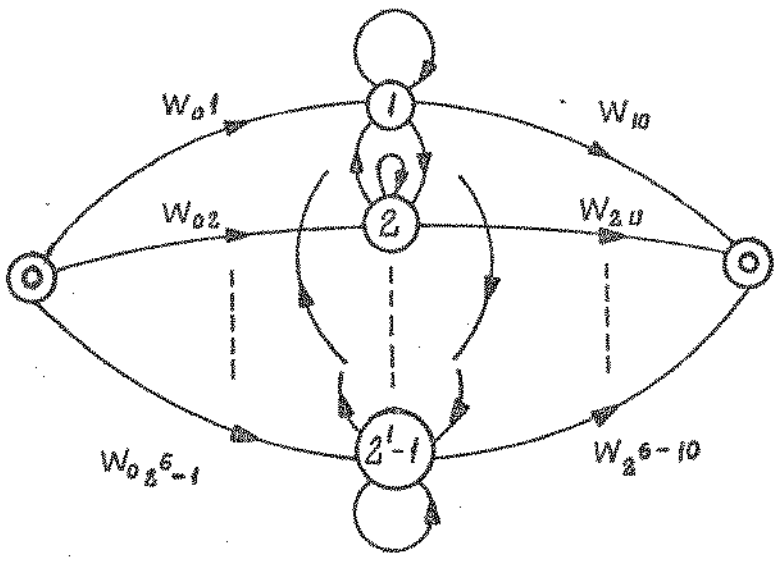


Figura 4.2.1 - Diagrama de estado particionado.

geração de uma classe de códigos de memória-unitária linear onde cada código nesta classe pode ser representado no diagrama de estado particionado como mostrado na Fig. 4.2.1.

Seja $\underline{d}$ a representação lexicográfica dos pesos de Hamming w_{oi} e w_{io} dos ramos que saem do estado zero para o estado i e do estado i para o estado zero, respectivamente, isto é,

$$\underline{d} = \{d_1, d_2, \ldots, d_{q^b-1}\}$$

com

$$d_i = w_{oi} + w_{io}, \qquad 1 \leqslant i \leqslant q^b-1$$

O problema que aqui defrontamos é o de encontrar cada um dos valores de w_{oi} e w_{io}. Consequentemente, se soubermos os valores de w_{oi} e w_{io} permitirá a determinação das matrizes G_0 e G_1 respectivamente. Deste modo, o único parâmetro que teremos que saber é a distância livre do código dado que em geral $d_i = d_\infty + i - 1$ para $1 \leqslant i \leqslant q-1$. Assim, os Teoremas do Capítulo 2 que versam sobre os limitantes superiores da distância livre do código convolucional e de memória-unitária serão de extrema importância na determinação de $\underline{d}$. Note que $\underline{d}$ assim determinado só pode ser encarado como um possível candidato.

Para facilidade de exposição, reproduzimos aqui esses limitantes superiores

$$d_{min} \leqslant \min_{p \geqslant 1} \{(q-1) \cdot (q^{b-1}/q^b - 1) \cdot (n/b) \cdot (p + b \cdot \nu)\} \tag{4.2}$$

e

$$d_{min} \leqslant \{(q-1) \cdot (q^{b-1}/q^b - 1) \cdot n\} \cdot (\nu + 1) \tag{4.3}$$

Devemos enfatizar que o menor valor do limitante de (4.2) e (4.3) é

a distância mínima irrestrita, respectivamente, isto é, d_∞. Uma vez que o conjunto $\underline{d}$ é lexicográfico, temos que $d_1 = d_\infty$ e que em geral $d_i = d_\infty + i - 1$ para todo i entre 1 e q - 1 e onde d_∞ é obtido diretamente de (4,2) ou (4.3) dependendo do caso em consideração.

Da propriedade de máximo fluxo, equação (4.1), e da Fig. 4.2.1, temos que

$$\sum_{i=1}^{q^b-1} a_i \cdot d_i = (\nu + 1) \cdot \Phi \qquad (4.4)$$

onde $\nu = 1$ para códigos de memória-unitária e a_i representa o número de vezes que o valor d_i aparece.

Usando uma representação vetorial para os elementos de (4.4), temos

$$\underline{a} * \underline{d} = \bar{\Phi} \qquad (4.5)$$

onde $*$ representa produto escalar.

Para um código de memória-unitária linear com taxa $r = b/n$ fixada, os valores de máximo fluxo Φ e do conjunto $\underline{d}$ são facilmente conhecidos através do emprego das equações (4.1) e (4.2) ou (4.3), respectivamente. Portanto, resolver (4.5) implica na determinação do vetor $\underline{a}$. Inquestionavelmente, a equação (4.5) é a caracterização matemática do Problema da Mochila. Portanto, a determinação de bons códigos de memória-unitária reduz-se à resolução do knapsack (4.5).

Da solução de (4.5) podemos através da representação modular de códigos linear de bloco verificar se os pesos de Hamming das palavras do código, d_i, com os seus respectivos números de palavras, a_i, são pertinentes a uma dada

matriz geradora. Em caso positivo, obtivemos então as sub matrizes geradoras G_o e G_1. Portanto, resolver (4.5) para códigos de memória-unitária linear onde b assume valores grandes, é equivalente a determinar entre no máximo q^b possíveis sub conjuntos de soluções aquelas que fornecerão os códigos desejados. Esta forma de dependência exponencial com relação ao comprimento dos dados de entrada dos possíveis sub conjuntos de soluções é a complexidade computacional respectiva ao melhor dos algoritmos conhecidos para resolver o problema da mochila.

Esses sub conjuntos são definidos como

$$A_i = \{(w_{o1}, w_{o2}, \ldots, w_{oq^b-1}), (w_{1o}, w_{2o}, \ldots, w_{q^b-1o})\}$$

onde w_{oj} e w_{jo} são os pesos de Hamming saindo do estado zero e indo para o estado j e os pesos de Hamming saindo do estado j e indo para o estado zero, respectivamente, para i entre 1 e q^b. Note que em cada sub conjunto existem $(q^b - 1)$ códigos equivalentes.

Como uma forma de explicitar o resultado que acabamos de estabelecer, apresentaremos um exemplo onde a solução do problema da mochila é trivial. Para isso, consideraremos um código de memória-unitária sobre GF(2) com taxa r = 2/4. Este código é equivalente a um código convolucional com memória $\nu = 2$ e taxa r = 1/2. Assim, determinação de um bom código de memória-unitária é equivalente a resolver a equação (4.5) onde as incógnitas são os a_i's. Isto é,

$$a_1 \cdot d_1 + a_2 \cdot d_2 + a_3 \cdot d_3 = 16 \qquad (4.6)$$

Da equação (4.2) ou (4.3), este código tem distância livre $d_\infty = 5$ e assim $d_1 = 5$, $d_2 = 6$ e $d_3 = 7$. Obviamente, a solução de (4.6) é $a_1 = 2$, $a_2 = 1$ e $a_3 = 0$.

Os quatro possíveis conjuntos de soluções os quais satisfazem (4.6) são $A_1 = \{(4,3,1), (1,3,4)\}$, $A_2 = \{(4,2,2), (1,3,4)\}$, $A_3 = \{(4,2,2), (2,3,3)\}$ e $A_4 = \{(2,3,3), (3,2,3)\}$. Embora os sub conjuntos A_3 e A_4 resultem em códigos com a mesma distância livre, A_4 é o ótimo. Portanto, uma possível solução para G_o e G_1 é dada por

$$G_o = \begin{bmatrix} 0 & 1 & 0 & 1 \\ & & & \\ 1 & 0 & 1 & 1 \end{bmatrix} \qquad G_1 \begin{bmatrix} 0 & 1 & 1 & 1 \\ & & & \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

onde "1" implica que existe uma conexão de um registro com o somador mod 2 e "0" o contrário. Como o codificador está sendo representado pelo paralelismo de 2 registros, temos que as primeiras e segundas linhas da sub matriz geradora $G_o(G_1)$ representam as conexões entre o primeiro (segundo) estágio dos registros no arranjo paralelo aos somadores mod 2.

4.3 - SISTEMAS CRIPTOGRÁFICOS UTILIZANDO CÓDIGOS DE TRELIÇA

Na Seção anterior, vimos que a solução do problema determinação de códigos de memória-unitária é reduzido à resolução do problema da mochila. Embora seja de amplo conhecimento que este problema pertence à classe de problemas conhecidos como Não Polinomiais Completos (NP-completo), no seu pior caso, isto viabiliza à proposição de sistemas criptográficos convencionais bem como de chave pública.

A idéia de empregar códigos corretores de erros em criptografia foi apresentada por McEliece. Neste esquema McEliece propos a utilização de códigos

de bloco, mais especificamente códigos de Goppa, com a finalidade de aplicação em sistemas de chave pública. Esta proposição, se baseia no fato de que os códigos de Goppa com capacidade de correção de t erros é difícil de ser quebrado, pois sua alta eficiência de correção é destruída se os bits que compõe as palavras do código sofrerem um embaralhamento antes da transmissão.

Além desse cripto-sistema, outros foram propostos e analisados. Esses cripto-sistemas estão sustentados em conceitos de Teoria dos Números e Teoria Combinatorial, onde um sem número de problemas reconhecidamente de difícil solução são utilizados como elementos estruturais na proposição de novos esquemas.

Os cripto-sistemas a serem apresentados nas sub seções seguintes serão denotados por sistemas convencionais lineares e não lineares, dependendo exclusivamente do tipo de funções aplicadas aos conteúdos dos registros, bem como cripto-sistemas de chave pública utilizando códigos convolucionais.

Acreditamos que o grau de dificuldade em quebrar cripto-sistemas tendo como base os códigos convolucionais se deve aos seguintes fatos: 1) o embaralhamento dos bits que compõe a palavra do código faz com que a eficiência do código na correção de erros seja destruída; 2) o fato adicional de que a seqüência interdependente de problemas da mochila do tipo (0 - 1) tem que ser resolvida. Devemos salientar que este knapsack não é do tipo supercrescente como proposto por Merkle e Hellman. Desse modo, as análises criptográficas realizadas para o caso do esquema proposto por Merkle e Hellman não serão eficientes na quebra do sistema aqui em consideração.

Como o problema da mochila pertence à classe NP-completo é, portanto, muito desejável na aplicação em criptografia e que somente corresponderá aos anseios esperados se uma função armadilha puder ser encontrada. Esta função

consiste essencialmente da aplicação de matrizes "inversíveis" às sub matrizes geradoras do código convolucional de tal forma a transformá-las em sub matrizes de um novo código convolucional com capacidade de correção menor do que o código original. A transformação direta é facilmente determinada, porém, a complexidade na determinação inversa é muito grande. Isto é um requisito fundamental para a análise criptográfica a que o sistema deverá ser submetido.

Em geral, encontramos problemas da mochila que são facilmente solucionáveis e outros de enorme complexidade. Este fato vale para códigos convolucionais bem como para códigos de memória-unitária e códigos de treliça. Exemplos mostrando isso para códigos de memória-unitária foi desenvolvido na Seção anterior.

O limiar entre essas duas classificações de problemas da mochila está exponencialmente relacionado com o comprimento dos dados de entrada.

A taxa sem dimensão de códigos convolucionais é definida por $r = b/n$, onde b é o comprimento dos dados de entrada e n o comprimento das palavras de ramo, com b e n inteiros. Como os códigos convolucionais podem ser representados por máquinas de estados finito, o número de estados e o número de ramos saindo de cada um dos estados são dados por $2^{b.\nu}$ e 2^{b}, respectivamente.

Como vimos anteriormente, o número de possíveis soluções ótimas do problema da mochila para um dado código de memória-unitária é q^{b}, onde q é a ordem do corpo em consideração. O número de possíveis soluções quando não levamos em consideração a condição de ótimo é $q^{b(\nu+1)n}$. Acontece que várias dessas soluções levam à códigos equivalentes no sentido de que o conjunto d são iguais. Porém, isto não é necessariamente verdadeiro para o caso dos cripto-sistemas a serem propostos uma vez que os knapsacks inteiros são diferentes.

Uma dificuldade ainda persiste, isto se deve ao fato de que para pe-

quenos valores de b, q^b é relativamente pequeno e, portanto, sendo razoavelmente fácil de ser resolvido o problema da mochila. Por outro lado, para b grande q^b é extremamente grande e portanto de difícil solução. Agora, para aplicações práticas gostaríamos de ter uma solução relativamente fácil. Este impasse pode ser resolvido através da aplicação de transformações apropriadas às sub matrizes geradoras de tal forma que possamos resolver um knapsack relativamente fácil, determinamos o código ótimo a ser utilizado, em seguida aplicamos transformações apropriadas de tal modo que o comprimento das palavras de saída tenham um comprimento maior, menor ou igual que o do código ótimo.

No caso do comprimento ser maior, o objetivo desta transformação é exclusivamente de aumentar a dimensionalidade. A consequência direta é que as palavras de ramo do código apresentam agora uma distância de Hamming maior que as originais. Obviamente, isto só traria benefícios ao cripto-analista. Portanto, uma nova transformação preservando a dimensionalidade deverá ser aplicada de tal forma a diminuir consideravelmente a distância de Hamming.

4.3.1 - Sistemas Criptográficos Convencionais .

Em criptografia convencional, a chave a ser utilizada é escolhida a partir de um conjunto de chaves cuja cardinalidade é grande. Sem dúvida teremos que manter em segredo a chave selecionada, pois a mesma tanto servirá para a cifragem como a decifragem.

Consistente com esta premissa, a utilização de códigos convolucionais em sistemas criptográficos convencionais será assim possível uma vez que interpretemos como chave o estado inicial, isto é, o conteúdo dos ν registros seguido da condição de não introduzir um número de zeros de tal modo a forçar o

retorno ao estado zero.

Seja dado um código convolucional com taxa r = b/n, $\nu + 1$ registros e que o comprimento da sequência de dados seja L . b. A condição acima mencionada, decorre do fato que a maneira usual de utilização de códigos convolucionais condiciona que os registros sejam fixados com valores iniciais zero (ou qualquer outro estado conhecido) e que após a uma sequência de dados de comprimento L . b dígitos tenham sido codificados, b . ν zeros sejam introduzidos novamente de modo a terminar a operação no estado zero ou naquele estado previamente fixado. Isto implica que independentemente da introdução de erros no canal à sequência codificada, existirá um caminho na treliça que divergirá do estado zero (ou qualquer outro estado conhecido) e retornará ao estado zero (ou ao estado conhecido) após (L + ν) ramos.

Isto deixa de ser verdadeiro para o caso quando b . ν zeros não são concatenados à sequência de dados de entrada mesmo que o estado inicial seja conhecido. Pois deixará de existir dígitos de controle de tal modo a impor a volta ao estado fixado. Desse modo o decodificador deverá levar em consideração todos os possíveis caminhos que conduzam do estado inicial à todos os estados terminais após L ramos da treliça. Por outro lado, se o estado inicial não é conhecido, o decodificador terá que levar em consideração também todos os estados iniciais.

Neste caso os últimos dígitos da sequência de entrada não terão a mesma proteção quando b . ν zeros são concatenados. Uma maneira de evitar isto é através da inserção nos registros os b dígitos iniciais ou terminais da sequência com comprimento L + b bits e em seguida iniciar o processo de codificação dos L + b dígitos da sequência de dados. Note que os estados iniciais e terminais são desconhecidos toda vez que L + b dados de entrada estão prontos para

serem codificados. De qualquer modo o decodificador deverá ter que realizar $2^{b.\nu}$ computações usando o algoritmo de Viterbi ou qualquer um dos algoritmos de decodificação Sequencial na decodificação de uma dada sequência de dados de entrada.

No caso dos códigos convolucionais lineares as operações sendo realizadas nos dígitos da sequência a ser transmitida é linear, mod 2. Para os códigos de memória-unitária não lineares não necessariamente isto é verdadeiro, pois essas operações são arbitrárias.

A diferença entre códigos de memória-unitária lineares e não lineares, com ou sem a concatenação de zeros e com ou sem o conhecimento do estado inicial, é que o decodificador para o caso de códigos não lineares terá de qualquer maneira que realizar comparações dos caminhos na treliça aos pares independentemente se o estado inicial é conhecido ou não. O motivo dessa comparação é intrínsico ao processo de decodificação. Este fato adiciona novas dificuldades ao cripto-analista uma vez que todas essas comparações terão que ser obrigatóriamente realizadas.

Esta análise é feita de modo eficiente usando-se o conceito de super estado. Porém, sua complexidade cresce com o quadrado do número de estados da treliça original. Agora, é de amplo conhecimento que este problema tem solução em tempo polinomial se todas as métricas acumuladas são iguais ou todos os comprimentos são iguais mesmo que o número de comparações seja grande. Se essas condições não são verdadeiras então este problema é do tipo NP-completo pois o mesmo é equivalente ao problema mínimo caminho com restrição de pesos. Consequentemente, temos que a segurança deste esquema está fundamentada no fato de que são problemas considerados "difíceis".

4.3.2 - Sistemas Criptográficos de Chave Pública

A seguir iremos apresentar alguns esquemas criptográficos de chave pública que utilizam os códigos convolucionais como os elementos geradores da cifragem e os algoritmos de Viterbi e Sequencial para a "decifragem".

4.3.2.1 - Knapsack binário

O cripto-sistema do tipo knapsack tendo como base os códigos convolucionais pode ser descrito como mostrado na Fig. 4.3.2.1.1.

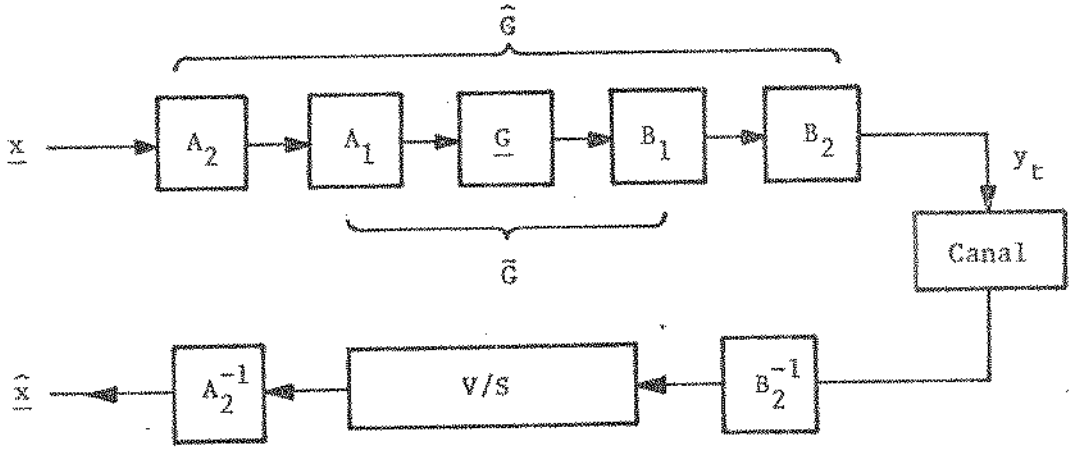


Figura 4.3.2.1.1 - Modelo do cripto-sistema de chave pública.

Na Figura acima, a caixa G representa na forma matricial a matriz ge

radora de um código convolucional geral. As caixas A e B representam transformações armadilhas apropriadas p x b e n x q, respectivamente, dando origem a um novo código convolucional com taxa r = p/q e com o mesmo número de registros que o código gerado por G. Dado os propósitos pelos quais essa transformação foi aplicada, ou seja, de conduzir à um código convolucional com capacidade de correção maior que o original, segue-se a aplicação de uma função armadilha consistindo de transformações $\tilde{A}$ e $\tilde{B}$ que são matrizes quadradas p x p e q x q respectivamente de modo a reduzir a capacidade de correção do novo código convolucional.

No receptor $\tilde{B}^{-1}$ é aplicado à sequência de saída do canal, antes de entrar na caixa rotulada V/S, que é a decodificação MLSE ou Sequencial, de modo a reduzir a complexidade computacional de decodificação. Em seguida a decodificação propríamente dita é realizada. Finalmente a transformação inversa de embaralhamento ($\tilde{A}^{-1}$) é aplicada.

Seja $\underline{x} = (\ldots, x_{-1}, x_0, x_1, \ldots)$ a sequência de dados de entrada onde cada x_i possue comprimento p. Cada sub matriz de G, G_i, tem dimensão b x n. Então a matriz A tem que ter dimensão p x b e B n x q. Como a determinação das sub matrizes passa pela solução do problema da mochila e que a complexidade deste problema é 2^b, então valores relativamente pequenos de b tem que ser usados. Suponha que esse problema tenha sido resolvido. Seguindo o modelo da Fig. 4.3.2.1.1, a aplicação das transformações A e B às sub matrizes geradoras do código convolucional como sendo uma função armadilha apropriada. Na Seção seguinte iremos provar que a eficiência do código com relação à correção de erros é destruída através do embaralhamento dos dígitos que compõe a sequência de dados.

Seja G_i com i entre 0 e ν, as sub matrizes geradoras do código convo

lucional. A aplicação das transformações A e B à G_i resulta em novas sub matrizes geradoras G_i' como segue

$$G_i' = A \cdot G_i \cdot B \qquad i = 0, 1, 2, \ldots, \nu$$

Note que a matriz A possue inversa a esquerda $\bar{A}$, se e somente se rank(A) = b. Ela possuirá inversa à direita se e somente se rank(A) = p. Similarmente, B terá inversa à esquerda se e somente se rank(B) = q e inversa à direita se e somente se rank(B) = n.

Como o código gerado por G' tem capacidade de correção maior que G, dado que essas transformações assim o permitem, teremos necessariamente que aplicar uma nova transformação à G' de modo a reduzir essa capacidade de correção à níveis desejados. Seja a função armadilha $\tilde{A}$ e $\tilde{B}$ matrizes inversíveis p x p e q x q, respectivamente. Então a nova função geradora $\tilde{G}_i$ será dada por

$$\tilde{G}_i = \tilde{A} \cdot G_i' \cdot \tilde{B} \qquad i = 0, 1, 2, \ldots, \nu$$

Como o sistema criptográfico é de chave pública, essas matrizes geradoras serão colocadas numa lista como

$$\tilde{G} = \left[\tilde{G}_o \; ; \; \tilde{G}_1 \; ; \; \ldots \; ; \; \tilde{G}_\nu\right]$$

Agora o processo de codificação de códigos convolucionais é definido por

$$y_t = x_t \cdot \tilde{G}_o + x_{t-1} \cdot \tilde{G}_1 + \ldots + x_{t-\nu+1} \cdot \tilde{G}_{\nu-1} + x_{t-\nu} \cdot \tilde{G}_\nu \qquad (4.7)$$

com $t \geq 0$ e $x_{-1} = \underline{0}$.

Poderemos ainda, se quisermos, adicionar um padrão de erro de igual valor da capacidade de correção de erro do código em consideração à y_t.

Como no caso mais geral $\tilde{A}$ e $\tilde{B}$ podem ser matrizes retangulares $pl \times p$ e $q \times ql$, respectivamente, suas inversas à direita e à esquerda A^+, B^+ e A^- e B^-, devem estar disponíveis no receptor. Essas inversas são dadas por

$$C^+ = C^t \cdot (C \cdot C^t)^{-1} \qquad \text{e} \qquad C^- = (C^t \cdot C)^{-1} \cdot C^t \tag{4.8}$$

com $C = A$ ou B, tal que $A^- \cdot \tilde{G}_i \cdot B^+ = G'_i$, onde t significa transposta.

A sequência y_t passa através do canal que poderá ou não introduzir erro(s). A saída do canal será a sequência de dígitos a ser decodificada. Antes da decodificação propriamente dita, aplicaremos $\tilde{B}^+$ à sequência de saída do canal de forma a obter

$$y_t \cdot \tilde{B}^+ = (x_t \cdot \tilde{A}) \cdot G'_0 + (x_{t-1} \cdot \tilde{A}) \cdot G'_1 + \ldots + (x_{t-\nu} \cdot \tilde{A}) \cdot G'_\nu$$

Após a aplicação dessa transformação à sequência de saída do canal, o processo de decodificação segue seu procedimento usual. À sequência decodificada aplicaremos A^+ com o objetivo de desembaralhar e produzir a sequência original.

Uma observação deve ser colocada em destaque, ou seja, se for desejável que o código intermediário tenha uma capacidade de correção de erros maior tem-se que um maior fator de multiplicidade entre p e q, com p constante, deverá ser utilizado; ou aumentando p quando $\nu = 1$; ou aumentando ν.

4.3.2.2 - Knapsack inteiro

Dado algumas limitações inerentes ao knapsack binário, que ficarão

evidentes a seguir, é que advém a proposição do knapsack inteiro. A essência desse novo esquema é a mesma que a do caso tratado anteriormente, onde ao invés de utilizarmos a representação binária dos elementos das sub matrizes geradoras, G_i ou G_i' para i entre 0 e ν, estaremos empregando números inteiros.

Da mesma forma que o knapsack na forma de bloco implementa o esquema proposto por Diffie Hellman, nesse novo esquema teremos um knapsack também na forma de bloco, porém, esses sub blocos irão ser utilizados em uma forma convolucional. Este fato implica na introdução de memória no processo de cifragem, consequentemente, colocando uma interdependência entre os sub blocos codificados a serem transmitidos.

Vamos supor que o knapsack binário tenha sido resolvido, isto é, dispomos das sub matrizes geradoras G_i'. Como mencionado anteriormente, iremos representar cada linha dessas sub matrizes geradoras por seu número inteiro correspondente. Este knapsack será "camuflado" através da aplicação da transformação s mod w, onde s e w são inteiros tal que o máximo divisor comum, gcd, é um, isto é, gcd(s,w) = 1. Aplicando a transformação s mod w à G_i', obteremos

$$\hat{G}_i = [(G_i' \cdot s) \bmod w] \qquad i = 0, 1, 2, \ldots, \nu$$

Como no knapsack binário, essas sub matrizes são colocadas em uma lista como

$$\hat{G} = [\hat{G}_0 ; \hat{G}_1 ; \ldots ; \hat{G}_\nu]$$

O processo de codificação é o mesmo que o da equação (4.7), onde o sinal + significa a operação usual de adição.

Antes de realizar o processo de decodificação, aplicaremos r mod w à

y_t, onde $r = s^{-1}$, de modo a obter

$$\hat{y}_t = (y_t \cdot r) \bmod w = x_t \cdot \left[(\hat{G}_o \cdot r) \bmod w\right] + \ldots + x_{t-\nu} \cdot \left[(\hat{G}_\nu \cdot r) \bmod w\right]$$

Em seguida, aplicaremos B^+ para obter

$$\hat{y}_t \cdot B^+ = (x_t \cdot A) \cdot G_o + \ldots + (x_{t-\nu} \cdot A) \cdot G_\nu$$

O resultado do processo de decodificação será $x_t \cdot A$. Com isso, o próximo passo será a aplicação de A^+ para a obtenção de x_t.

Devemos salientar que: 1) o knapsack convolucional não é do tipo super crescente; 2) a segurança e consequentemente a complexidade desse sistema está fundamentada nos argumentos de necessidade de um número muito grande de computações na determinação das pseudo-inversas bem como para a solução do problema da mochila.

4.3.2.3 - Cripto-sistema RSA convolucional de chave pública

A variação que passaremos a descrever, segue a idéia de operar diretamente de forma convolucional com o knapsack binário ou inteiro das linhas de $\hat{G}_i$.

Seja k um número inteiro. Então, para cada k p-bloco de dados de entrada, $p \cdot \nu$ zeros são concatenados e um padrão de erro estabelecido é adicionado aos correspondentes k q-blocos codificados. Cada um desses k q-blocos são então transformados em números inteiros e "camuflados" através da transformação s mod w. Note que mais de uma transformação pode ser aplicada.

Essas operações são suficientes para a segurança e privacidade das mensagens a serem transmitidas. Contudo, desejamos melhorar o processamento da informação no último estágio através do emprego do sistema criptográfico RSA de chave pública. Como resultado desses passos, teremos um sistema criptográfico RSA convolucional de chave pública em contra posição ao RSA de bloco.

Esta variação será descrita após a introdução sumária do cripto-sistema RSA. No sistema RSA são selecionados dois números primos $\tilde{p}$ e $\tilde{q}$ e definindo $\tilde{A}$ como $\tilde{A} = \tilde{p} \cdot \tilde{q}$. Uma vez que conheçamos a função Totiente de Euler, isto é, $\phi(\tilde{A}) = (\tilde{p} - 1) \cdot (\tilde{q} - 1)$, selecionaremos um número inteiro E entre 2 e $\phi(\tilde{A})$ de modo a não ter nenhum fator comum com $\phi(\tilde{A})$. Em seguida, determina-se um número inteiro D que seja o "inverso" de E mod $\phi(\tilde{A})$. Em seguida E e $\tilde{A}$ são publicados numa lista.

A cifragem de uma mensagem M, representada pelo correspondente texto cifrado C, a qual pode ser colocada como um número inteiro entre 0 e $\tilde{A} - 1$ segue a operação $C = M^E \bmod \tilde{A}$. O processo de decifragem do texto cifrado C segue através do emprego do número de decifragem D através da operação $M = C^D \bmod \tilde{A}$.

Neste ponto, estamos aptos a introduzir a seguinte variação: sejam dados E, D, $\tilde{A}$ e $\left[\hat{G}_o ; \hat{G}_1 ; \ldots ; \hat{G}_\nu\right]$, o cifrador, o decifrador, um inteiro, e as sub matrizes geradoras do código convolucional com taxa r = b/n.

O sistema RSA convolucional invariante no tempo consiste basicamente na publicação em uma lista dos elementos E, $\tilde{A}$ e $\left[\hat{G}_o ; \hat{G}_1 ; \ldots ; \hat{G}_\nu\right]$. Qualquer um que desejar se comunicar com o usuário identificado por esses três elementos, assim o fará através da codificação de p dígitos da sequência de dados de entrada através da matriz geradora publicada. A saída codificada será então transformada para o inteiro correspondente. Seja M_i tal inteiro. Este número será cifrado através do uso do sistema RSA, isto é, por

$$C_i = (M_i)^E \bmod \tilde{A}$$

Note que para uma dada sequência de dados de comprimento N . p estará sendo codificada convolucionalmente em uma sequência de comprimento N e portanto a ela estará sendo associado um caminho na treliça. Decodificação dessa sequência de comprimento N segue através do algoritmo de Viterbi ou Sequencial onde cada um desses N inteiros é processado a priori através do sistema RSA, isto é, $M_i = (C_i)^D \bmod \tilde{A}$.

4.4 - PROPRIEDADES DAS FUNÇÕES ARMADILHAS

Nesta Seção estaremos estabelecendo as propriedades das funções armadilhas com relação às condições de otimalidades dos códigos a serem empregados nos sistemas criptográficos tratados na Seção anterior.

Sejam as sub matrizes geradoras G_i com dimensão b x n e onde $0 \leq i \leq \nu$, de um código de memória-unitária sobre o corpo de dois elementos, GF(2). Seja $\phi(G_i)$ o peso total de Hamming de cada G_i . $\phi(G_i)$ é dada explicitamente por

$$\phi(G_i) = n \cdot 2^{b-1}, \qquad 0 \leq i \leq \nu$$

Definição : Uma matriz G com dimensão b x n sobre GF(2) é dita ser igualmente distribuída com relação aos pesos de Hamming se e somente se os $2^b - 1$ elementos não nulos gerados por G tem pesos de Hamming

$$\tilde{w}_H = (G)/(2^b - 1).$$

Note que se $2^b - 1$ dividir $\phi(G)$ então $[\tilde{w}] = \tilde{w}$, caso contrário $[\tilde{w}] = \{|\tilde{w}|, |\tilde{w}| \pm j, \ j \geq 0\}$. Como os códigos de memória-unitária são equivalentes aos códigos convolucionais, iremos considerar somente os de memória-unitária onde G a matriz geradora tem dimensão $2b \times n$. Deste modo, a busca por códigos convolucionais ótimos a serem utilizados nos sistemas criptográficos será restrita à:

Lema : Se o rank(G) = 2b e $2b \leq n$, então G tem 2b vetores linhas linearmente independentes com pesos de Hamming igualmente distribuídos e são capazes de gerar todas as 2^{2b} palavras do código também com pesos de Hamming igualmente distribuídos.

Demonstração : Um código de memória-unitária com taxa r = b/n tem sua representação em treliça com 2^b estados e transições de cada estado. Deste modo o número total de transições será 2^{2b}. Agora, como 2b vetores linha de G são linearmente independentes, G gera um espaço vetorial com 2^{2b} vetores.
Seja $[\tilde{w}]$ a parte inteira de $\tilde{w}$. Existem $n!/(n - [\tilde{w}])! \, . \, [\tilde{w}]!$ vetores tendo peso de Hamming igual a $[\tilde{w}]$. Como $n \geq 2b$, pode-se mostrar que $n!/(n - [\tilde{w}])! \, . \, [\tilde{w}]! \geq 2b$. Assim, pelo menos 2b vetores tem peso de Hamming $[\tilde{w}]$. Como cada transição tem associado um vetor diferente, é possível encontrar um código de bloco (b, 2n) com o tamanho do bloco igual a 2n e o tamanho da informação b tal que a matriz geradora seja constituída de vetores tendo pesos de Hamming igualmente distribuídos com distância mínima igual à do código de bloco (b, 2n). Esta é a menor das distâncias, uma vez que outras transições poderão ser maiores de pelo menos uma unidade.

CQD

A consequência imediata deste Lema é que o código é do tipo não catastrófico. Por outro lado, se $2b > n$ a base do espaço vetorial gerado por G tem dimensão $\dim(G) \leq n$, então pelo menos uma das linhas de G é uma combinação linear das demais. Isto implica que nem todas transições tem associado um vetor diferente. Então, a busca pelo código ótimo deve ser bastante criteriosa, uma vez que o mesmo pode ser catastrófico.

Uma vez que o código ótimo tenha sido determinado e supondo que qualquer uma das funções armadilhas (A_1, B_1) ou (A_2, B_2) que representaremos por (A, B), é aplicada às sub matrizes geradoras do código com taxa $r = b/n$ e $\nu + 1$ registros. Estas transformações resultam em novos códigos com a mesma taxa, e então teremos

Proposição 1 : Seja G_i, $0 \leq i \leq \nu$, sub matrizes geradoras de um código ótimo de memória-unitária não catastrófico sobre GF(2). Sejam A e B matrizes inversíveis com dimensões $b \times b$ e $n \times n$, respectivamente, então

$$d_{min}\{A \cdot [G_o, G_1, \ldots, G_\nu] \cdot B\} \leq d_{min}\{[G_o, G_1, \ldots, G_\nu]\}$$

onde A e B não são necessariamente unitárias.

Demonstração : Por hipótese temos que $G = [G_o, G_1, \ldots, G_\nu]$ é a matriz geradora de um código ótimo não catastrófico. Seja $d_{min}(G) = d_{min}$. Agora, seja $G' = A.G.B$ com $d_{min}(G') = d'_{min}$. Com isso teremos as seguintes condições: 1) $d'_{min} > d_{min}$ ou 2) $d'_{min} \leq d_{min}$.

Definiremos S como o conjunto de todas as possíveis transformações (A, B) sobre GF(2). Note que S pode ser particionado em três sub conjuntos S_1, S_2 e S_3 tal

que S_1 é o conjunto de transformações (A, B) que conduzem a códigos com d_{min} maiores, S_2 é o conjunto de transformações (A, B) que conduzem aos códigos equivalentes e S_3 é o conjunto de todas as outras transformações (A, B) que conduzem aos códigos com menores d_{min}.

Se a condição 1) é verdadeira, isto implica que existe uma transformação (A, B) em S_1 que resulta em um código com distância mínima ainda maior. Tal código somente poderá ser catastrófico. Caso contrário estará contradizendo a hipótese de que o código original era ótimo e não catastrófico. Desse modo a condição 1) não é satisfeita.

Agora, G' é equivalente a G se $G' = A.G.B$ onde A e B são matrizes inversíveis com dimensão b x b e n x n com determinantes iguais a 1. Então, (A, B) pertence à S_2 e a igualdade é válida em 2). Obviamente, se A e B pertencem à S_3, então a transformação resultante conduzirá a um código com capacidade corretora de erros menor. CQD

Exemplo da Proposição 1 : Seja um código convolucional com taxa $r = 2/4$ e $\nu = 1$. Um bom código convolucional no conjunto de todos os códigos convolucionais com taxa $r = 2/4$ e $\nu = 1$ pode ser determinado ao resolver-se o problema da mochila correspondente às suas sub matrizes. Uma das soluções gera um código com sub matrizes dadas por

$$G_o = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \qquad G_1 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

A distância mínima deste código é $d_{min}(2/4) = 5$. Sem perda de generalidades, seja a transformação (A_2, B_2) composta por matrizes quadradas dadas por

$$A_2 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad B_2 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Aplicando essa transformação em G_o e G_1, teremos G_o' e G_1' dadas por

$$G_o' = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \qquad G_1' = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

A distância mínima deste novo código é $d_{min}(2/4) = 3$. Deste modo, a eficiência de correção do código foi destruída, pois agora somente um erro pode ser corrigido, enquanto o código original corrigia até dois erros.

Uma outra condição é possível quando temos que o código foi determinado segundo as condições do Lema e que os conjuntos de funções armadilhas (A_1, B_1) e (A_2, B_2) são aplicadas às sub matrizes geradoras do código com taxa $r = b/n$ e número de registros $\nu + 1$. Estas aplicações resultam em novos códigos com o mesmo número de registros, porém, com taxa $r = b/q$. Deste modo teremos

Proposição 2 : Seja G_i, $0 \leq i \leq \nu$, sub matrizes geradoras de um código ótimo de memória-unitária não catastrófico sobre GF(2). Seja $d_{min}(b/n)$ a distância mínima deste código, isto é,

$$d_{min}\{[G_o, G_1, \ldots, G_\nu]\} = d_{min}(b/n)$$

Existem matrizes A e B com dimensões $b \times b$ e $n \times q$, respectivamente, com $q > n$ sobre GF(2) transformando G_i em $\tilde{G}_i$, novas sub matrizes $b \times q$ com

$$d_{min}\{[\tilde{G}_o, \tilde{G}_1, \ldots, \tilde{G}_\nu]\} = d_{min}(b/q)$$

tal que $d_{min}(b/n) \leq d_{min}(b/q)$.

Demonstração : Seja $d_{min}(b, n)$ a distância mínima de um código de bloco (b, n) com tamanho das palavras do código n e dos bits de informação b. Não temos dúvida que $d_{min}(b/n) = d_{min}(b, n)$. A partir do Lema, temos que $d_{min}(b,n) \leq d_{min}(b,2n)$ e que se q > n então $d_{min}(b, 2q) \geq d_{min}(b, 2n)$. Assim, $d_\infty(b/q) \geq d_\infty(b/n)$. Note que a igualdade é válida se G e G' são equivalentes onde det(A) = 1 e rank(B^+) = n. CQD

Exemplo da Proposição 2 : Seja um código convolucional com taxa r = 2/3 e $\nu = 1$. A solução do problema da mochila relativo à este código nos leva às seguintes sub matrizes geradoras G_o e G_1 dadas por

$$G_o = \begin{bmatrix} 0 & 1 & 1 \\ & & \\ 1 & 1 & 0 \end{bmatrix} \qquad G_1 = \begin{bmatrix} 0 & 1 & 0 \\ & & \\ 1 & 0 & 1 \end{bmatrix}$$

A distância mínima deste código é $d_{min}(2/3) = 3$. Sem perdas de generalidades, seja a transformação (A, B) composta por matrizes quadrada e retangular, respectivamente dadas por

$$A = A^+ = \begin{bmatrix} 1 & 0 \\ & \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad B^+ = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Aplicando esta transformação em G_o e G_1, teremos novas sub matrizes G_o' e G_1'

dadas por

$$G_o' = \begin{bmatrix} 0 & 1 & 1 & 0 \\ & & & \\ 1 & 0 & 1 & 1 \end{bmatrix} \qquad G_1' = \begin{bmatrix} 0 & 1 & 1 & 1 \\ & & & \\ 1 & 1 & 0 & 1 \end{bmatrix}$$

A distância mínima deste novo código é $d_{min}(2/4) = 5$. Portanto, $d_{min}(2/3) \leq d_{min}(2/4)$.

Como uma generalização da Proposição 2, onde as transformações (A_1, B_1) e (A_2, B_2) são tais que conduzam à novos códigos com taxa $r = p/q$ e com o mesmo número de registros que o código original, temos

Proposição 3 : Seja G_i, $0 \leq i \leq \nu$, sub matrizes geradoras de um código ótimo de memória-unitária não catastrófico sobre GF(2). Seja $d_{min}(b/n)$ a distância mínima deste código, isto é,

$$d_{min}\{[G_o, G_1, \ldots, G_\nu]\} = d_{min}(b/n)$$

Existem matrizes A e B com dimensões $p \times b$ e $n \times q$, respectivamente, com $p \geq b$ e $q > n + 1$ sobre GF(2) transformando G_i em $\tilde{G}_i$, novas sub matrizes $p \times q$ com

$$d_{min}\{[\tilde{G}_o, \tilde{G}_1, \ldots, \tilde{G}_\nu]\} = d_{min}(p/q)$$

tal que $d_{min}(b/n) \leq d_{min}(p/q)$.

Demonstração : Segue do Lema, da demonstração da Proposição 2 e do limitante superior da distância mínima dado por

$$d_{min}(b/n) \leq \min_{u \geq 1} \{[2^{u-1}/(2^u - 1)] \cdot (n/b) \cdot (u + b \cdot \nu)\}$$

Note que G' é equivalente à G se o rank(A^-) = b, rank(A^+) = p e rank(B^+) = n

CQD

Através dessas três Proposições conseguimos estabelecer condições de existência das funções armadilhas. Essas propriedades estão ligadas ao aspecto de segurança desses cripto-sistemas quanto à sua complexidade computacional.

4.5 - ANÁLISE DE COMPLEXIDADE COMPUTACIONAL

É de conhecimento geral que o processo ótimo para a decodificação de códigos convolucionais é o algoritmo de Viterbi. Entretanto, devido à limitações de implementação este algoritmo torna-se viável para códigos com um número de registros relativamente pequeno, enquanto que a alternativa para valores grandes de números de registros é a decodificação Sequencial.

Por outro lado, $2^{b.\nu}$ é o parâmetro limitante na seleção da técnica de decodificação a ser utilizada. Como os códigos de memória-unitária, em geral, possuem pelo menos a mesma distância mínima que a dos códigos convolucionais específicos para um mesmo número de estados e sabendo que esses códigos são equivalentes, podemos então dizer que a complexidade e portanto a segurança desses cripto-sistemas dependem exponencialmente do comprimento dos dados de entrada.

Com isso, estamos aptos a estabelecer a complexidade dos cripto-sistemas de chave pública do tipo knapsack usando os códigos convolucionais com as

respectivas funções armadilhas.

A complexidade quando usamos a decodificação de Viterbi será definida como sendo o número de operações por ramo. Na lista pública, encontraremos $p \times 1$ sub matrizes geradoras G_i, $0 \leq i \leq \nu$, sobre os inteiros positivos, Z^+. Para um cripto-analista decodificar uma mensagem, é necessário construir uma tabela contendo todas as possíveis combinações lineares dos $p(\nu+1)$ elementos do knapsack. O número de operações necessárias é obtido como se segue: para cada valor de ramo sendo decidificado (p bits) uma pré-computação consiste de até $p \cdot \nu$ adições seguida de uma comparação. No j-ésimo intervalo de tempo p novos bits terão que ser decodificados. Deste modo teremos no máximo p adições, resultando em $p(\nu+1)$ operações. Como o alfabeto de entrada pertence ao GF(2), o número total de operações será $2^{p(\nu+1)}$. Assim a complexidade é $o(2^{p(\nu+1)})$.

Agora, relembre que havíamos aplicado algumas transformações, funções armadilhas $\tilde{A}$ e $\tilde{B}$ às $\nu+1$ sub matrizes geradoras G_i' do código convolucional. Assim, de modo a recuperar a sequência de dados originais o cripto-analista terá que dispor das pseudo inversas de $\tilde{A}$ e $\tilde{B}$.

No caso mais geral, as transformações $\tilde{A}$ e $\tilde{B}$ serão matrizes retangulares com dimensões $pl \times p$ e $q \times ql$, respectivamente. O processo de decodificação e decifragem exigem o conhecimento das respectivas pseudos inversas. Essas pseudos inversas são dadas pela equação (4.8), a qual reescrevemos aqui para facilidade de exposição, isto é,

$$C^- = (C^t \cdot C)^{-1} \cdot C^t \qquad \text{e} \qquad C^t = C^t \cdot (C \cdot C^t)^{-1} \cdot C^t$$

se e somente se $\text{rank}(A^-) = p$, $\text{rank}(A^+) = pl$, $\text{rank}(B^-) = ql$ e $\text{rank}(B^+) = q$, respectivamente, onde t significa transposta.

Quando as matrizes $\tilde{A}$ e $\tilde{B}$ são quadradas, o número de computações para determinarmos $\tilde{A}^{-1}$ e $\tilde{B}^{-1}$ é $(p^3 + q^3)$. Quando elas são retangulares, temos que $\tilde{A}^t \cdot \tilde{A}$ e $\tilde{B} \cdot \tilde{B}^t$ são matrizes simétricas necessitando pois de p^3 e q^3 computações, respectivamente. Multiplicação por $\tilde{A}^t$ e $\tilde{B}^t$ necessita $ql \cdot q^2$ e $pl \cdot p^2$ computações, respectivamente. Assim, o número total de computações quando $\tilde{A}$ e $\tilde{B}$ são matrizes retangulares é $(pl \cdot p^5 + ql \cdot q^5)$.

Consequentemente, a complexidade final destes sistemas criptográficos propostos usando das funções armadilhas e decodificação de Viterbi ou Sequencial é $o(2^{p(\nu+1)} \cdot [pl \cdot p^5 + ql \cdot q^5])$ quando as transformações são matrizes retangulares e $o(2^{p(\nu+1)} \cdot [p^3 + q^3])$ quando as matrizes são quadradas.

Se por outro lado, desejarmos ter uma estimativa quanto ao erro de bit que o sistema de comunicações estará sujeito na utilização desses sistemas criptográficos será suficiente levar em consideração o argumento da função $o(\cdot)$. Para tal, representemos esse argumento por χ. Pode-se demonstrar que, para grandes valores de χ e para taxas em bits/símbolo $R_o < R < C$, a probabilidade de erro de bit é assintoticamente dada por

$$P_b \doteq \chi^{-\rho}$$

onde

$$E_c(R)/R = \rho, \qquad 0 < \rho \leq 1$$

com $E_c(\cdot)$ sendo a função de Gallager para códigos convolucionais.

Com relação à decodificação Sequencial, esta análise em termos de P_b versus a quantidade computacional também pode ser feita. Para isso, definiremos como complexidade da decodificação Sequencial o número máximo de computações, $\tilde{C}_i$, no sub conjunto incorreto de cada nó normalizado por p. Mais especificamen-

te, consideraremos o número de armazenadores necessários de modo a controlar o fluxo.

Seja B . q o número de símbolos de entrada do canal quando a saída do mesmo é quantizada em M níveis. O número de bits a serem armazenados é dado por $B \cdot q \cdot \lceil \log_2 M \rceil$. Seja u o número de computações efetuadas por ramo durante um intervalo de chegada dos bits dos ramos dividido por p bits. Então, $\tilde{C}_i > \mu \cdot B$ corresponde ao número de computações necessárias no i-ésimo sub conjunto incorreto.

Pode-se mostrar que a probabilidade de "overflow" é limitada inferiormente por

$$P_{overflow} > (\mu \cdot B)^{-\rho} \cdot [1 - o(\mu \cdot B)]$$

onde

$$E_o(\rho)/\rho = R, \qquad 0 < \rho < \infty \quad e \quad 0 < R < C$$

com $E_o(\cdot)$ sendo a função de Gallager.

Portanto, $\mu \cdot B$ tem o mesmo significado que $2^{p(\nu+1)}$ na decodificação de Viterbi.

Referências

L.N. Lee, "Short unit-memory byte-oriented binary convolutional codes having maximal free distance," *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 349-352, Maio 1975.

W. Diffie and M.E. Hellman, "New direction in cryptography," *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 644-654, Nov. 1976.

R.C. Merkle, "Secure communication over an insecure channel," *Common. Assoc. Mach.*, vol. 21, pp. 294-299, Abr. 1978.

R.C. Merkle and M.E. Hellman, "Hiding information and signatures in trapdoor knapsacks," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 525-530, Set. 1978.

R.L. Rivest, A. Shamir and L. Adleman, "On digital signatures and public key cryptosystems," *Common. Assoc. Comput. Mach.*, vol. 21, pp. 120-126, Fev. 1978.

Y.G. Desmedt, J.P. Vandewalle and R.J.M. Govaerts, "A critical analysis of the security of knapsack public key algorithms," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 601-610, Julho 1984.

A. Shamir, "A polynomial time algorithm for breaking the basic Merkle-Hellman cryptosystem," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 699-704, Set. 1984.

D. Coppersmith, "Fast evaluation of algorithms in fileds of characteristic two," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 587-593, Julho 1984.

A.M. Odlyzko, "Cryptanalytic attacks on the multiplicative knapsack cryptosystem and on Shamir's fast signature shceme," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 594-600, Julho 1984.

R.J. McEliece, "A public key system based on algebraic coding theory," *JPL DSN* Progress Rep., 1978.

R. Palazzo Jr., "Cryptographic systems based on trellis codes," *3rd. Brazilian Symposium on Telecommunications*, São José dos Campos, Brazil, Set. 2-4, 1985.

R. Palazzo Jr., "Linear unit-memory codes - a knapsack problem?," *2nd Swedish-USSR International Workshop on Information Theory*, Granna, Sweden, Abril 14-19, 1985.

R. Palazzo Jr., "New short constraint length convolutional codes derived from a network flow approach," *IEEE Intern. Symp. on Inform. Theory*, Brighton, England, Junho 23-28, 1985.

R. Palazzo Jr., "Unit-Memory Convolutional Codes as a Means in Cryptographic Systems," *2nd SIAM Conference on Applied Linear Algebra*, Raleigh, North Carolina, Abril 29 - Maio 2, 1985.

M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Co., San Francisco, 1979.

A.J. Viterbi and J.K. Omura, *Principles of Digital Communications and Coding*, McGraw-Hill, 1979.

R. Palazzo Jr., "Propriedades das Funções Armadilhas e Análise de Complexidade de Cripto-Sistemas de Chave Pública Usando Códigos Convolucionais", *5º Simpósio Brasileiro de Telecomunicações*, Campinas, Setembro 8-10, 1987. (submetido)

R. Palazzo Jr., M.C.P. Young e R.C.F. Cruz, "On the knapsack Combinatorial Problem of Unit-Memory Codes", *International Symposium on Information and Coding Theory*, Campinas, Julho 27-Agosto 1, 1987. (submetido)

R. Palazzo Jr., "Public Key Cryptosystems Based on Trellis Codes", *IEEE Transaction on Information Theory*, submetido para publicação.

CAPÍTULO 5

CÓDIGOS DE TRELIÇA NÃO LINEARES CICLO-ESTACIONÁRIOS

5.1 - INTRODUÇÃO

Todos os códigos apresentados e analisados nos capítulos anteriores formam uma sub-classe dos códigos de treliça não lineares. Como nos casos anteriores, podemos ter códigos de treliça não lineares invariantes e variantes no tempo.

Dado que os sistemas de comunicações considerados *mais eficientes*, em geral, são não lineares e variantes no tempo, é que apresentaremos neste capítulo uma análise matemática baseada numa medida de distorção média sobre sequências ciclo-estacionárias geradas por automatas de estados finitos do tipo Mealey.

A política de otimização, via programação dinâmica, utiliza de uma métrica geral e do limitante de Chernoff de modo a fornecer um limitante superior da medida de distorção média bem acurado.

Este limitante superior é apresentado na forma matricial para facilidade de cálculos.

Um outro aspecto relevante sobre a análise a ser apresentada está relacionado com o grau de liberdade na escolha da medida de distorção a ser utilizada. Dependendo do critério desejado para a avaliação do desempenho do sistema uma escolha conveniente da função "medida de distorção" será necessária. Dessa forma, a avaliação de sistemas dinâmicos gerais pode ser obtida.

Para ser mais direto, através da medida de distorção média estamos procurando estabelecer um procedimento sistemático de tal forma que políticas de decisão ótimas, via programação dinâmica, possam ser enumeradas através de um critério pré-estabelecido.

Este processo de contagem encontra aplicação, por exemplo em codifi-

cação para controle de erros. Em geral, a maior distância mínima de um código é o critério de otimalidade empregado. Entretanto, para sistemas de comunicações este critério é o de mínima probabilidade de erro de bit.

Como em geral, é extremamente difícil de se obter expressões analíticas para a probabilidade de erro de bit, então limitantes superiores e inferiores bem acurados são estabelecidos de modo a sobrepor tal dificuldade.

Considerando o exemplo acima mencionado como a motivação desejada, é de conhecimento que esquemas de codificação para controle de erros podem ser gerados por automatas de estados finitos (AEF). O AEF que estamos interessados é o do tipo Mealey que é descrito matematicamente por uma 5-upla (S, I, O, P, Q) onde S é o conjunto de estados, I é o conjunto de entradas, O é o conjunto de saídas e P e Q são funções tais que $P : I, S \rightarrow S$ e $Q : I, S \rightarrow O$.

Como estamos interessados em códigos não lineares, a função Q será arbitrária, possivelmente não linear, sobre o corpo de Galois, GF, com q elementos, GF(q).

O AEF em consideração será descrito por b entradas paralelas cada contendo ν memórias e funções arbitrárias F e G ε Q a serem aplicadas ao conteúdo das memórias.

Note que a estrutura da treliça, sendo gerada pelo AEF do tipo Mealey, é descrita pela função P, enquanto que o valor associado à cada transição é dado através de F e G ε Q. Esta treliça consiste de q^b ramos saindo e chegando em cada estado e de $q^{b\nu}$ estados.

Seja $\underline{\alpha} = (\alpha_1, \alpha_2, \alpha_3, \ldots)$ a sequência de dados de entrada onde cada α_i tem comprimento b. Através das funções P e Q aplicadas em $\underline{\alpha}$ temos que a treliça fica completamente caracterizada em sua estrutura e valor, respectiva-

mente, e que a cada $\underline{\alpha}$ teremos associado um correspondente caminho na treliça. Como cada caminho na treliça pode ser caracterizado por uma sequência de estados, temos assim que a cada $\underline{\alpha}$ temos uma sequência de estados associados Θ, $(P(\underline{\alpha}) = \Theta)$.

Sejam $\underline{x}$ e $\underline{x}'$ duas sequências quaisquer de saída com comprimento N, dadas por

$$\underline{x} = (x_{u+1}, x_{u+2}, \ldots, x_{u+N})$$

$$\underline{x}' = (x'_{u+1}, x'_{u+2}, \ldots, x'_{u+N})$$

onde x_i e x'_i tem comprimento n e $u \in N^*$. Note que $\underline{x}$ e $\underline{x}'$ podem ser vistos como duas possíveis políticas com comprimento n . N. Associaremos com a sequência $\underline{x}$ a política "correta" a ser seguida e com a sequência $\underline{x}'$ a política "incorreta".

Podemos interpretar $\underline{x}'$ como sendo uma sequência que divergiu da sequência $\underline{x}$ no instante $t = u + 1$ em algum estado pertencente a S_k e convergiu posteriormente no instante $t = u + N$ no estado S_j, com k e $j \in \{1, 2, 3, \ldots, M\}$ ou de M é o número de estados.

Definiremos $m(\underline{x}, \underline{y})$ como a métrica a ser utilizada no processo de otimização. Podemos visualizar $m(\cdot, \cdot)$ como uma função "casamento", isto é, $m(\cdot, \cdot)$ possue a propriedade de escolher $\underline{x}$ tão próxima de $\underline{y}$ quanto desejarmos. Seja $\Gamma(\underline{x}, \underline{x}'; \underline{y})$ a métrica diferença, isto é, $\Gamma(\underline{x}, \underline{x}'; \underline{y}) = m(\underline{x}', \underline{y}) - m(\underline{x}, \underline{y})$.

Como F e G são funções arbitrárias, $\Gamma(\underline{x}, \underline{x}'; \underline{y})$ pode ou não ter valores acumulados diferentes para cada $\underline{x}$, $\underline{x}'$ e $\underline{y}$ quando o estado inicial S_k e final S_j assumem os valores do conjunto de estados.

Assim, o objetivo deste capítulo é de apresentar uma técnica que per

mitirá medir quão boa é a política ótima. Na Seção 5.2 limitantes superior e inferior serão apresentados. Na Seção 5.3 aplicação destes limitantes para o problema de resposta parcial periodicamente variante no tempo será apresentado. Na Seção 5.4 a aplicação na avaliação do desempenho de modulação por sobreposição será estudada. Finalmente, na Seção 5.5 uma aplicação para a avaliação do desempenho de sistemas de comunicações de múltiplo acesso utilizando de um esquema combinado de "Polling"/Códigos Convolucionais é analisado.

5.2 - LIMITANTES DA MEDIDA DE DISTORÇÃO MÉDIA

Iremos apresentar nesta seção os limitantes superiores e inferiores da medida de distorção média na forma matricial para facilidade de cálculos numéricos.

Inicialmente, assumiremos que os AEF, em número de dois, geram sequências ciclo-estacionárias com período 2. Em seguida uma generalização para N-AEFs com período qualquer será feita.

Suponha que à cada b símbolos que entram nos AEFs n símbolos são produzidos como saída nos instantes de tempo $t = 2j$ e $t = 2j + 1$ relativos aos AEF I e AEF II, respectivamente. Matematicamente temos que se $\underline{\alpha} = (\alpha_1, \alpha_2, \ldots)$ é a sequência de dados e $\underline{x}_1$ e $\underline{x}_2$ são as saídas dos AEFs I e II, então

$$\underline{x}_1 = F(\alpha_i) = (x_{11}^i, x_{12}^i, \ldots, x_{1n}^i)$$

$$\underline{x}_2 = G(\alpha_j) = (x_{21}^j, x_{22}^j, \ldots, x_{2n}^j)$$

Seja $d[\gamma', (\tilde{s}_\gamma, \tilde{\alpha}_\gamma), (s_\gamma, \alpha_\gamma)]$ a medida de distorção com $\gamma' = \gamma \bmod W$ onde W é a periodicidade em consideração. Note que para cada valor γ', $d[\gamma', \cdot, \cdot]$ pode representar funções diferentes.

A princípio estamos interessados em funções do tipo *característica*

$$d[\gamma', (\tilde{s}_\gamma, \tilde{\alpha}_\gamma), (s_\gamma, \alpha_\gamma)] = \begin{cases} 1, & \tilde{\alpha}_\gamma \neq \alpha_\gamma \\ 0, & \tilde{\alpha}_\gamma = \alpha_\gamma \end{cases}$$

e do tipo *erro quadrático*

$$d[\gamma', (\tilde{s}_\gamma, \tilde{\alpha}_\gamma), (\tilde{s}_\gamma, \alpha_\gamma)] = \begin{cases} (\tilde{\alpha}_\gamma - \alpha_\gamma)^2, & \tilde{\alpha}_\gamma \neq \alpha_\gamma \\ 0, & \tilde{\alpha}_\gamma = \alpha_\gamma \end{cases}$$

Seja $(s_\gamma, \alpha_\gamma)$ os verdadeiros valores do estado e do dado de entrada e $(\tilde{s}_\gamma, \tilde{\alpha}_\gamma)$ o estado e dado de entrada selecionados pelo algoritmo da política ótima no intante $t = \gamma$.

Um evento de erro é caracterizado por $(\tilde{s}_\gamma, \tilde{\alpha}_\gamma) \neq (s_\gamma, \alpha_\gamma)$ onde $u < \gamma < v$, isto é, que o algoritmo da política ótima eliminará um caminho onde $(s_\gamma, \alpha_\gamma)$ pertence, caminho este que iniciou em $t = u$ e terminará em $t = v$. Como o processo é ciclo-estacionário para cada instante discreto de tempo podemos assumir diferentes métricas.

Seja $m(\gamma', x_\gamma, y_\gamma)$ uma métrica definida por

$$m(\gamma', \underline{x}, \underline{y}) = \beta \ln \{ p(\underline{y}/\underline{x}, \gamma') \} + w$$

$$= \sum_{\gamma=u}^{v} \beta \ln \{ p(y_\gamma / x_\gamma, \gamma') \} + w$$

onde β e w são constantes, $\ln$ é o logarítimo natural, e $p(\cdot / \cdot, \cdot)$ é a função densidade de probabilidade condicional.

Seja $\{(s_\gamma, \alpha_\gamma)\}$ a sequência verdadeira do estado e dado de entrada. Suponha que $\{(\tilde{s}_\gamma, \tilde{\alpha}_\gamma)\}$ é uma possível sequência de estado onde:

$$s_u = \tilde{s}_u , \quad s_v = \tilde{s}_v \quad \text{e} \quad s_k \neq \tilde{s}_k \quad \text{para} \quad u < k < v$$

e

$$\sum_{k=u}^{v-1} m\left[k', (\tilde{s}_k, \alpha_k), y_k\right] \geq \sum_{k=u}^{v-1} m\left[k', (s_k, \alpha_k), y_k\right]$$

Sejam $\underline{s}(u, v)$, $\underline{\tilde{s}}(u, v)$ e ρ', tal que

$$\underline{s}(u, v) = (s_u, s_{u+1}, \ldots, s_v)$$

$$\underline{\tilde{s}}(u, v) = (\tilde{s}_u, \tilde{s}_{u+1}, \ldots, \tilde{s}_v)$$

$$\rho' = u \bmod w$$

A probabilidade de que a sequência $\underline{\tilde{s}}(u, v)$ será escolhida em relação à verdadeira sequência $\underline{s}(u, v)$ é dada por

$$\Pr\left[\underline{s}(u, v) \neq \underline{\tilde{s}}(u, v)\right] = \Pr\left[\sum_{k=u}^{v-1} m(k', (\tilde{s}_k, \alpha_k), y_k \geq \sum_{k=u}^{v-1} m(k', (s_k, \alpha_k), y_k)\right] \tag{5.1}$$

Para que (5.1) seja verdade, $y_k = x_k + n_k$ com $\{n_k\}$ variáveis aleatórias independentes. Usando o limitante de Chernoff, com $\lambda \geq 0$, temos

$$\Pr\left[\underline{s}(u, v) \to \underline{\tilde{s}}(u, v) / \rho'\right] \leq$$

$$= E\left\{ \exp\left[\sum_{k=u}^{v-1} \lambda(m\left[k', (\tilde{s}_k, \tilde{\alpha}_k), y_k\right] - m\left[k', (s_k, \alpha_k), y_k\right]) / \underline{s}, \underline{\tilde{s}}, \rho' \right] \right\}$$

$$= \prod_{k=u}^{v-1} E\left\{ \exp\left[\lambda(m\left[k', (\tilde{s}_k, \tilde{\alpha}_k), y_k\right] - m\left[k', (s_k, \alpha_k), y_k\right])\right] / \underline{s}, \underline{\tilde{s}}, \rho' \right\}$$

$$= \prod_{k=u}^{v-1} D\left[\lambda, k', (\tilde{s}_k, \tilde{\alpha}_k), (s_k, \alpha_k), \rho'\right] \qquad (5.2)$$

Note que (5.2) limita superiormente a probabilidade de evento de erro específico, isto é, um dado evento de erro. Considere agora o conjunto de todos os eventos de erro que iniciaram no instante $t = u$ e terminaram no instante $t = v$.

Seja $S_{u,v}(\underline{s}(u, v))$ o conjunto de todas as subsequências de erro, isto é,

$$S_{u,v}(\underline{s}(u, v)) = \{\underline{\tilde{s}}(u, v) \ : \ s_u = \tilde{s}_u, \ s_v = \tilde{s}_v \ \text{ e } \ s_k \neq \tilde{s}_k, \quad u < k < v\}$$

A distorção média condicional entre a sequência de métrica máxima $\{(s'_k, \alpha'_k)\}$ e a verdadeira $\{(s_k, \alpha_k)\}$ no instante $t = n$ é limitada por

$$E\{d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] / \underline{s}, \rho'\} \leq$$

$$\sum_{n\geq u} \sum_{n<v} \sum_{S_{u,v}} d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] \cdot \Pr\left[\underline{s}'(u, v) = \underline{\tilde{s}}(u, v) / \underline{s}(u, v)\right]$$

$$\leq \sum_{n>u} \sum_{n<v} \sum_{S_{u,v}} d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] \cdot \Pr\left[\underline{s}(u, v) \rightarrow \underline{\tilde{s}}(u, v)\right]$$

Esta última desigualdade vem do fato de que a probabilidade que $\underline{\tilde{s}}(u, v)$ tenha a métrica máxima entre todas as subsequências de eventos de erro é menor do que a probabilidade que $\underline{\tilde{s}}(u, v)$ tenha a maior métrica do que aquela da sequência verdadeira $\underline{s}(u, v)$.

A distorção média descondicionada com relação ao estado é dada por

$$E\left\{ d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] / \rho' \right\} =$$

$$= \sum_{\underline{s}(u,v)} E\left\{ d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] / \underline{s}(u, v), \rho' \right\} \cdot \Pr\left[\underline{s}(u, v)\right] \qquad (5.3)$$

A subsequência verdadeira $\underline{s}(u, v)$ representa uma cadeia de Markov de primeira-ordem, uma vez que as equações de estado e saída são dadas por

$$x_k = Q(s_k, \alpha_k)$$

$$s_{k+1} = P(s_k, \alpha_k)$$

Deste modo,

$$\Pr\left[\underline{s}(u, v)\right] = \Pr\left[s_v / s_{v-1}\right] \cdot \Pr\left[s_{v-1} / s_{v-2}\right] \cdot \ldots \cdot \Pr\left[s_{u+1} / s_u\right] \cdot \Pr\left[s_u\right] =$$

$$= \Pr\left[\alpha_{v-1}\right] \cdot \Pr\left[s_{v-2}\right] \cdot \ldots \cdot \Pr\left[\alpha_u\right] \cdot \Pr\left[s_u\right] \qquad (5.4)$$

onde $Pr[s_u]$ é a probabilidade de estado estacionário de s_u.

Substituindo (5.4) em (5.3), e fazendo algumas manipulações algébricas, temos

$$E\left\{ d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] / \rho' \right\} \leq$$

$$\sum_{n \geq u} \sum_{n < v} \sum_{S_{u,v}} d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] . Pr\left[\underline{s}(u, v) \rightarrow \underline{\tilde{s}}(u, v)\right] .$$

$$Pr\left[\underline{s}(u, v)\right] \qquad (5.5)$$

Substituindo (5.2) em (5.5), temos

$$E\left\{ d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] / \rho' \right\} \leq$$

$$\sum_{n \geq u} \sum_{n < v} \sum_{S_{u,v}} d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] . Pr\left[s_u\right] .$$

$$. \prod_{k=u}^{v-1} Pr(\alpha_k) . D\left[\lambda, k', (\tilde{s}_k, \alpha_k), (s_k, \alpha_k), \rho'\right] .$$

Note que neste caso o processo todo não é estacionário, uma vez que para qualquer deslocamento de comprimento L, temos que $Pr[\underline{s}(u, v)] = Pr[\underline{s}(u + L, v + L)]$ mas $Pr[\underline{s}(u, v) \rightarrow \underline{\tilde{s}}(u, v)]$ pode ou não ser igual a $Pr[\underline{s}(u + L, v + L) \rightarrow \underline{\tilde{s}}(u + L, v + L)]$. Por outro lado, para valores de L tal que $L = k . W$, com W o período da combinação e k um número natural, a igualdade vale para qualquer $\rho' = u \bmod W$ sendo o início do evento de erro. Assim, temos que o processo é ciclo-estacionário e consequentemente

$$E\left\{d\left[n', (\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)\right] / \rho'\right\} \leq$$

$$\sum_{q=1}^{\infty} \sum_{S_o,q} \left\{ \sum_{r=0}^{q-1} d\left[r', (\tilde{s}_r, \alpha_r), (s_r, \alpha_r)\right] \right\} \cdot Pr\left[s_o\right] \cdot$$

$$\cdot \prod_{k=0}^{q-1} Pr\left[\alpha_k\right] \cdot D\left[\lambda, k', (\tilde{s}_k, \tilde{\alpha}_k), (s_k, \alpha_k), \rho'\right] \qquad (5.6)$$

Definindo $\tilde{d}_{\rho'}$, como a média estatística da medida de distorção condicionada por ρ' e $\delta = \frac{d}{dz} z^{\delta}\Big|_{z=1}$ e substituindo em (5.6) temos

$$\tilde{d}_{\rho'} \leq \frac{d}{dz} T_{\rho'}(z)\Big|_{z=1}$$

onde $$T_{\rho'}(z) = \sum_{q=1}^{\infty} \sum_{S_o,q} Pr\left[s_o\right] \cdot \prod_{r=0}^{q-1} z^{d\left[r', (\tilde{s}_r, \tilde{\alpha}_r), (s_r, \alpha_r)\right]} \cdot$$

$$\cdot Pr(\alpha_r) \cdot D\left[\lambda, r', (\tilde{s}_r, \tilde{\alpha}_r), (s_r, \alpha_r), \rho'\right] \qquad (5.7)$$

Seja $\tilde{d}$ "ensemble" da distorção média, isto é

$$\tilde{d} = \sum_{\rho'} \tilde{d}_{\rho'} \, Pr\left[\rho'\right]$$

Como cada AEF tem a mesma chance de iniciar o processo, então

$$\tilde{d} = \left(\frac{1}{2}\right)\left[\tilde{d}_1 + \tilde{d}_2\right]$$

assim

$$\tilde{d} \leq \left(\frac{1}{2}\right) \frac{d}{dz}\{T_1(z) + T_2(z)\} = \left(\frac{1}{2}\right) \frac{d}{dz} T(z)\Big|_{z=1}$$

Todos os casos em que o critério de máxima verossimilhança é emprega do $\tilde{d}$ pode ser reduzido de um fator 1/2. Deste modo,

$$\tilde{d} \leq \frac{1}{4} \frac{d}{dz} T(z)\Big|_{z=1}$$

Note que $[(\tilde{s}_n, \tilde{\alpha}_n), (s_n, \alpha_n)]$ representa estados $(\tilde{s}_n, s_n)$ e dados de entrada $(\tilde{\alpha}_n, \alpha_n)$. Assim, definindo $W_n = (\tilde{s}_n, s_n)$ e $A_n = (\tilde{\alpha}_n, \alpha_n)$ como super-estados e super-entradas temos

$$\tilde{W} = \{W_{n,m} = (\tilde{s}_n, s_m) , \quad 0 \leq n, \ m \leq M\}$$

$$\tilde{A} = \{A_{n,m} = (\tilde{\alpha}_n, \alpha_m) , \quad 0 \leq n, \ m \leq M\}$$

onde M é a cardinalidade dos conjuntos de estado e dados de entrada. Note também que $\tilde{W}$ é composto de dois subconjuntos disjuntos $\tilde{W}_e$ e $\tilde{W}_{ne}$ dados por

$$\tilde{W}_e = \{(\tilde{s}_n, s_m) , \quad 0 \leq n, \ m < M \ : \ n = m \ \text{e} \ \tilde{s}_n = s_m\}$$

$$\tilde{W}_{ne} = \{(\tilde{s}_n, s_m) , \quad 0 < n, \ m < M \ : \ n \neq m \ \text{e} \ \tilde{s}_n \neq s_m\}$$

cujas cardinalidades são M e $M^2 - M$ respectivamente.

Defina

$$\Pr\left[\tilde{W}_o\right] = \Pr\left[S_o\right]$$

$$d(k', \tilde{W}_k, \tilde{A}_k) = d\left[k', (\tilde{s}_k, \tilde{\alpha}_k), (s_k, \alpha_k)\right]$$

$$\Pr\left[\tilde{A}_k\right] = \Pr\left[\alpha_k\right]$$

$$D\left[\lambda, k', \tilde{W}_k, \tilde{A}_k, \rho'\right] = D\left[\lambda, k', (\tilde{s}_k, \alpha_k), (s_k, \alpha_k), \rho'\right]$$

Substituindo estas definições em (5.7), temos

$$T(z) = \sum_{m=1}^{\infty} \sum_{S_{o,m}} \Pr\left[\tilde{W}_o\right] \cdot \prod_{k=0}^{m-1} z^{d(k', \tilde{W}_k, \tilde{A}_k)} \cdot \Pr\left[\tilde{A}_k\right] \cdot D\left[\lambda, k' \tilde{W}_k, \tilde{A}_k, \rho = 0\right] +$$

$$+ \sum_{q=1}^{\infty} \sum_{S_{o,q}} \Pr\left[\tilde{W}_o\right] \cdot \prod_{j=0}^{q-1} z^{d(j, \tilde{W}_j, \tilde{A}_j)} \cdot \Pr\left[\tilde{A}_j\right] \cdot D\left[\lambda, j', \tilde{W}_j, \tilde{A}_j, \rho = 1\right]$$

onde $\rho = 0$ significa que o AEF I está operando e $\rho = 1$ o AEF II.

Defina H^k para $k = 1, 2$ como

$$H^k = \begin{bmatrix} h^k_{M+1,M+1} & h^k_{M+2,M+1} & \cdots & h^k_{M^2,M+1} \\ h^k_{M+1,M+2} & h^k_{M+2,M+2} & \cdots & h^k_{M^2,M+2} \\ \vdots & \vdots & & \vdots \\ h^k_{M+1,M^2} & h^k_{M+2,M^2} & \cdots & h^k_{M^2,M^2} \end{bmatrix}$$

onde

$$h_{ij}^{k} = \begin{cases} z^{d(k,\tilde{W}_i,\tilde{A}_i)} \cdot Pr\left[\tilde{A}_i\right] \cdot D\left[\tilde{A}, k, \tilde{W}_i, \tilde{A}_i\right], & \text{se } \tilde{W}_i \rightarrow \tilde{W}_j \\ 0, & \text{se } \tilde{W}_i \not\rightarrow \tilde{W}_j \end{cases}$$

Defina B_i^k e C_i^k como sendo

$$B_i^k = \left[h_{i,M+1}^k \quad h_{i,M+2}^k \quad \cdots \quad h_{i,M^2}^k\right]^t$$

$$C_i^k = \left[h_{M+1,i}^k \quad h_{M+2,i}^k \quad \cdots \quad h_{M^2,i}^k\right]^t$$

para $1 \leq i \leq M$, $k = 1, 2$ onde t significa transposta e H^k é a matriz de transição do AEF k e B_i^k e C_i^k são matrizes das condições iniciais e de saída para o estado i do AEF k.

Na forma matricial temos que a função de transferência pode ser estabelecida uma vez que definamos $t_i(z)$ como a função de transferência a partir de todos os estados iniciais para o estado intermediário $\tilde{W}_{ne}$. Consequentemente, o vetor função de transferência intermediário é dado por

$$t^k(z) = \left[t_{M+1}(z) \quad t_{M+2}(z) \quad \cdots \quad t_{M^2}(z)\right]^t, \quad k = 1, 2$$

o qual satisfaz

$$t^1(z) = H^1 \cdot t^2(z) + \sum_{i=1}^{M} p(\delta_i)\, B_i^1 \qquad (5.8)$$

$$t^2(z) = H^2 \cdot t^1(z) + \sum_{i=1}^{M} p(\delta_i)\, B_i^2 \qquad (5.9)$$

com δ_i e $\tilde{W}_e$.

A solução de (5.8) e (5.9) é dada por

$$t^1(z) = (I - H^1 \cdot H^2)^{-1} \left[H^1 \cdot \left(\sum_{i=1}^{M} p(\delta_i) B_i^2 \right) + \sum_{i=1}^{M} p(\delta_i) \cdot B_i^1 \right] \qquad (5.10)$$

$$t^2(z) = H^2 (I - H^1 \cdot H^2)^{-1} \left[H^1 \left(\sum_{i=1}^{M} p(\delta_i) B_i^2 \right) + \sum_{i=1}^{M} p(\delta_i) B_i^1 \right] + \sum_{i=1}^{M} p(\delta_i) B_i^2 \qquad (5.11)$$

As equações de saída para cada j, $0 < j < M$, é dada por

$$T_j^1(z) = (C_j^1)^t(z) \cdot t^2(z) \qquad (5.12)$$

$$T_j^2(z) = (C_j^2)^t(z) \cdot t^1(z) \qquad (5.13)$$

e

$$T^k(z) = \sum_{j=1}^{M} T_j^k(z) \qquad (5.14)$$

Substituindo (5.12) e (5.3) em (5.14), temos

$$T^1(z) = \sum_{j=1}^{M} (C_j^1)^t(z) \cdot t^2(z)$$

$$T^2(z) = \sum_{j=1}^{M} (C_j^2)^t(z) \cdot t^1(z)$$

Como

$$T(z) = T^1(z) + T^2(z)$$

então

$$T(z) = \left[\sum_{j=1}^{M} (C_j^1)^t(z)\right] \cdot t^2(z) + \left[\sum_{j=1}^{M} (C_j^2)^t(z)\right] \cdot t^1(z)$$

Finalmente,

$$\bar{d} \leq \frac{1}{4} \frac{d}{dz} T(z)\Big|_{z=1}$$

A generalização para N AEFs e período N será dada pelo Lema 1 a seguir. Note que este Lema guarda uma estreita semelhança com aquele apresentado no Capítulo 3, porém, aqui temos uma generalização uma vez que estamos considerando uma medida de distorção.

Lema 1 : Para qualquer sequência ciclo-estacionária com período N gerada por N-AEF, a distorção média é limitada superiormente por

$$\bar{d} \leq \left(\frac{1}{2N}\right)\left[\sum_{u=1}^{N-1} (\bar{C}_{u+1} \cdot t^u + C_{u+1} \cdot \bar{t}^u) + \bar{C}_1 t^N + C_1 \bar{t}^N\right]$$

onde as barras significam derivadas com relação a z e

$$C_u = \sum_{j=1}^{M} (C_j^u)^t(z) \quad , \qquad t^u = t^u(z)$$

e

$$t^N = \left(I - \prod_{i=o}^{N-1} H_{N-i} \right)^{-1} \cdot \left(\sum_{\gamma=1}^{N-1} \left(\prod_{m=o}^{\gamma-1} H_{N-m} \right) t^{N-\gamma} + t^N \right)$$

$$t^k = \left(\prod_{i=o}^{k-1} H_{k-i} \right) \left(I - \prod_{i=o}^{N-1} H_{N-i} \right)^{-1} \cdot \left(\sum_{\gamma=1}^{N-1} \left(\prod_{m=o}^{\gamma-1} H_{N-m} \right) t^{N-\gamma} + t^N \right) +$$

$$+ \sum_{k=1}^{k-1} \left[\left(\prod_{i=o}^{k-n-1} H_{k-i} \right) t^n + t^k \right] \quad , \quad \text{para} \quad 1 \leqslant k \leqslant N-1$$

Demonstração : Seja t^i a matriz representando os estados intermediários com dimensão $(M^2 - M) \times 1$ do i-ésimo AEF, $1 \leqslant i \leqslant N$, cuja matriz de transição é H^i.

A evolução do processo é descrita matematicamente por

$$t' = B_1 + H^1 \cdot t^N$$

$$t^2 = B_2 + H^2 \, t'$$

$$\vdots$$

$$t^{N-1} = B_{N-1} + H^{N-1} \, t^{N-2}$$

$$t^N = B_N + H^N \cdot t^{N-1}$$

e

$$T_1 = C_1 \cdot t^N$$

$$T_2 = C_2 \, t'$$

$$\vdots$$

$$T_N = C_N \cdot t^{N-1}$$

com

$$T = \left(\frac{1}{N}\right)\left[\sum_{i=1}^{N} T_i\right]$$

Note que os AEF evoluem ciclicamente segundo o padrão $\{L_1, L_2, \ldots, L_N\}$. Resolvendo o conjunto de equações para t^k teremos obtido o valor de t^N. Com esse valor nas demais equações obteremos t^k para $1 \leq k \leq N-1$ C.Q.D.

O limitante inferior da média estatística da medida de distorção é dado por

$$\tilde{d} \geq \min\{\tilde{d}_i\}$$

onde $\tilde{d}_i$ é a distorção média do i-ésimo AEF na combinação.

5.3 - APLICAÇÃO NA AVALIAÇÃO DE SISTEMAS DE COMUNICAÇÕES COM SINAIS DE RESPOSTA PARCIAL

Nesta Seção uma aplicação da técnica desenvolvida na Seção 5.2 a um sistema de comunicações PAM não codificado com interferência entre símbolos controlada e que varia periodicamente no tempo será apresentada.

Iniciaremos esta apresentação através de um fato bastante conhecido de que se a conformação de pulsos do transmissor ao receptor é do tipo "cosseno levantado" então interferência entre símbolos é reduzida ou controlada. Esse controle advem do uso de duas técnicas, a saber a duobinária e a duobinária modificada. Por outro lado, o preço pago através do uso desses controles para diminuir a interferência entre símbolos está relacionado com o aumento da relação

sinal-ruído.

Sabendo disto, algumas questões surgem naturalmente, isto é, existe alguma outra técnica que resulte numa performance melhor do que aquela apresentada pela técnica duobinária? Se existir, que ganhos em termos de relação sinal-ruído são possíveis?

A resposta para a primeira pergunta é sim e a técnica a ser utilizada é aquela de combinações periódicas de sinais duobinários ou sinais duobinários modificados. Consequentemente a resposta à segunda pergunta dependerá da combinação per si.

Em geral, interferência entre símbolos invariantes no tempo tem sido a forma de interferência empregada nas análises. Receptores de máxima verossimilhança para sinais digitais com interferência intersimbólicas invariantes no tempo foram propostos utilizando estruturas de estados finitos. Como estas estruturas podem ser descritas por uma treliça, então as técnicas desenvolvidas nos capítulos anteriores bem como a da Seção 5.2 podem ser utilizadas para a análise destes sistemas de comunicações.

Primeiramente, iremos considerar o modelo do sistema de comunicações que iremos adotar. O canal é do tipo contínuo discreto no tempo. O alfabeto de entrada será binário. Assumiremos que cada símbolo é independente e identicamente distribuído. Cada símbolo da fonte, também binário, com duração T seg., alimentará o canal representado através das respostas ao impulso $h_1(t)$ e $h_2(t)$. A saída do canal é amostrada durante T seg. e adicionado uma amostra do processo gaussiano com média zero e densidade espectral dos dois lados igual a $\frac{N_o}{2}$.

Sem perda de generalidades, assumiremos que a estratégia de seleção na saída do canal é cíclica, isto é, entre todas as possíveis combinações das

respostas ao impulso do canal tomadas duas a duas selecionaremos aquela onde $h_1(t)$, $h_2(t)$, $h_1(t)$, $h_2(t)$, ... ocorre.

A saída da fonte é representada por

$$v(t) = \sum_n u_n \, \delta(t - nT)$$

Na saída do canal temos

$$x(t) = \sum_n u_n \, \delta(t - nT) * h_{\tilde{n}}(t) = \sum_n u_n \, h_{\tilde{n}}(t - nT) \qquad (5.15)$$

onde T é a duração do símbolo T, n é o n-ésimo símbolo da fonte, $\tilde{n} = n \bmod 2$, $*$ é a convolução e $u_n \in \{0, 1\}$.

Assumiremos que $h_{\tilde{n}}(t) = 0$ para $|t| \geq LT$ e $\tilde{n} = 0, 1$. Como $h_{\tilde{n},k} = h_{\tilde{n},-k}$ para $1 \leq k \leq L-1$, definiremos o comprimento de memória de cada $h_{\tilde{n}}(t)$ como $L-1$. Como existe memória, observações em um dado intervalo de tempo dependem das anteriores, portanto, teremos que usar receptor com memória. Dentre os receptores não lineares temos que o de máxima verossimilhança é o que minimiza a probabilidade de erro da sequência de dados e que, portanto, será o receptor a ser utilizado.

No receptor teremos

$$y(t) = x(t) + n(t)$$

Seja $x_m(t)$ o sinal transmitido. De (5.15) temos que

$$x_m(t) = \sum_n u_{nm}\, h_{\tilde{n}}(t - nT) \tag{5.16}$$

Como o receptor é o de máxima verossimilhança, este decidirá por $x_m(t)$ se

$$p(y(t) / x_m(t)) \geq p(y(t) / x_{m'}(t)) , \qquad \forall\, m' \neq m \tag{5.17}$$

Através da técnica de Gram-Schmidt para determinação de bases ortonormais, representaremos $y(t)$ e $x_m(t)$ por seus coeficientes $\underline{y}$ e $\underline{x}_m$.

Como o ruído é Gaussiano e Branco então

$$p(\underline{y} / \underline{x}_m) = \left(\frac{1}{\pi N_o}\right)^{N/2} \exp\left\{-\frac{|\underline{y} - \underline{x}_m|^2}{N_o}\right\} \tag{5.18}$$

Assim $\underline{x}_m$ será selecionado se e somente se

$$\begin{aligned} x_m(t) &= \max_{\underline{x}_m}{}^{-1} \left\{ \ell n\ p(\underline{y} / \underline{x}_m) \right\} \\ &= \max_{\underline{x}_m}{}^{-1} \left\{ \sum_n 2u_{nm}\, y_n - \sum_j u_{nm}\, u_{mj}\, h_{\tilde{n},n-j} \right\} \end{aligned} \tag{5.19}$$

onde

$$y_n = \int_{-\infty}^{\infty} h_{\tilde{n}}(t - nT)\, y(t)\, dt$$

$$h_{\tilde{n},n-k} \int_{-\infty}^{\infty} h_{\tilde{n}}(t - nT)\, h_{\tilde{k}}(t - kT)\, dt \tag{5.20}$$

Substituindo (5.20) no último termo de (5.19) teremos

$$\sum_{n}\sum_{j} u_n u_j h_{\tilde{n},n-j} = \sum_{n}\left[u_n^2 h_{\tilde{n},o} - 2\sum_{i=1}^{L-1} u_n u_{n-i} h_{\tilde{n},i}\right] \qquad (5.21)$$

Observe que de (5.20) temos que o receptor ótimo conterá um filtro casado com a resposta ao impulso do canal e que esta informação é fornecida, através de algum processo, ao receptor. Note que este é um caso ideal. Se a resposta ao impulso não é conhecida então técnicas adaptativas deverão ser utilizadas. Entretanto, o desempenho do sistema deverá ser no máximo igual ao desempenho quando o conhecimento da resposta ao impulso é precisamente conhecida.

Substituindo (5.21) em (5.19), e observando que o lado direito de (5.19) representa uma métrica, temos que esta métrica depende de y_n, u_n e dos $L-1$ dados de entrada. Note que esta métrica descreve o comportamento do sistema de uma maneira similar à aquela descrita pelo algoritmo de Viterbi, onde a treliça contem 2^{L-1} estados e os ramos são caracterizados pelas respostas ao impulso nos tempos apropriados.

Agora, multiplicando (5.16) por $h_{\tilde{k}}(t-kT)$, integrando e fazendo alguns arranjos algébricos, temos que a saída do canal y_k é dada por

$$y_k = x_{mk} + n_k = \sum_{i=-L+1}^{L-1} u_{m,k-i}\, h_{\tilde{k},i} + n_k$$

onde

$$n_k = \int_{-\infty}^{\infty} n(t)\, h_{\tilde{k}}(t-kT)dt$$

Note que a média estatística de n_k é nula e que $E\{n_k\, n_j\} = \frac{N_o}{2} h_{\tilde{k},k-j}$

A partir destas considerações passaremos então a analisar o sistema de comunicações descrito anteriormente quando no receptor conhecemos precisamente a resposta ao impulso do canal e assim estaremos usando um filtro casado. Sob essa hipótese denominaremos o receptor ótimo.

Para sua análise, seja $x_m(t)$ o sinal transmitido. Na treliça este sinal está associado a um caminho com N ramos de comprimento. Assim um erro ocorrerá, na treliça, se durante o intervalo de N ramos um caminho apresentar uma métrica acumulada $\phi_{m'}$, equação (5.19), maior do que a métrica acumulada ϕ_m do caminho correto. Deste modo, a probabilidade deste evento de erro é dada por

$$Pr\left[\underline{y}_m \to \underline{y}_{m'}\right] = Pr\left[\phi_{m'} > \phi_m / \underline{x}_m\right]$$

Pode-se mostrar que $\{\phi_m - \phi_{m'}\}$, é Gaussiana com média

$$K = \sum_n \left[2(u_{m'n} - u_{mn})\hat{y}_n - (u^2_{m'n} - u^2_{mn})h_{\tilde{n},o} - \right.$$
$$\left. - \sum_{i=1}^{L-1} 2(u_{m'n}\, u_{m',n-i} - u_{mn}\, u_{m,n-i})h_{\tilde{n},i} \right]$$

onde $\hat{y}_n = \sum_{i=-L+1}^{L-1} u_{m,n-i}\, h_{\tilde{n},i}$

e variância $\bar{W} = \left(\frac{1}{4\sigma^2}\right) \cdot E\left\{ \sum_n 2(u_{m'n} - u_{mn})n_n \right\}^2$

onde σ^2 é a variância das variáveis aleatórias Gaussianas.

Desta forma

$$\Pr\left[\phi_{m'} - \phi_m > 0 \,/\, \underline{x}_m\right] = Q(k/2\sigma\sqrt{\tilde{W}})$$

Como estamos usando filtro casado, pode-se mostrar que k = W. Assim,

$$\Pr\left[\phi_{m'} - \phi_m > 0 \,/\, \underline{x}_m\right] = Q(\sqrt{\tilde{W}}/2\sigma)$$

Seja F^* a função $F^* : \{0, 1\} \to \{-1, 1\}$. Seja ε_n o sinal de erro definido por:

$$\varepsilon_n = \left(\frac{1}{2}\right)(u_{m'n} - u_{mn}) = \begin{cases} 0 \;, & u_{m'n} = u_{mn} \\ \pm 1 & u_{m'n} \neq u_{mn} \end{cases} \qquad (5.23)$$

Substituindo (5.23) em K e $\tilde{W}$, usando o limitante $\exp\{-x^2/2\}$ para Q(x), após algumas manipulações algébricas em (5.22) teremos

$$\sqrt{\tilde{W}}/2\sigma = \left[(2/N_o)\sum_n \sum_j \varepsilon_n \varepsilon_j h_{\tilde{n},n-j}\right]^{1/2}$$

e

$$\Pr\left[\phi_{m'} - \phi_m > 0 \,/\, \underline{x}_m\right] \leq \exp\left\{\left(-\frac{1}{N_o}\right)\left(h_{\tilde{n},o}\,\varepsilon_n^2 + 2\sum_{i=1}^{L-1} \varepsilon_n \varepsilon_{n-i} h_{\tilde{n},i}\right)\right\} \qquad (5.24)$$

Como (5.24) caracteriza a probabilidade de erro condicionada em $\underline{x}_m$,

ou seja, a uma sequência de erro específica, fazendo-se a média estatística sobre todas as possíveis sequências de erros teremos

$$\Pr\left[\text{erro do sinal}\right] \leq \sum_{\underline{\varepsilon}} \left\{ \prod_n \left(\frac{z}{2}\right)^{\varepsilon_n^2} \cdot \exp\left[\left(-\frac{1}{N_o}\right)\left(h_{\tilde{n},o}\, \varepsilon_n^2 + 2 \sum_{i=1}^{L-1} \varepsilon_n\, \varepsilon_{n-i}\, h_{\tilde{n},i}\right)\right]\right\} \tag{5.25}$$

onde z leva em conta o caso em que $u_{m'n} \neq u_{mn}$.

Note que a expressão entre as chaves caracteriza caminhos no diagrama de estado onde erros são considerados. Este diagrama de estado contem 3^{L-1} estados. Cada caminho é o resultado do produto dos valores de ramos correspondentes às respostas de impulso dos canais em cada tempo apropriado. Desta forma o limitante superior de (5.25) é a função de transferência dos 2 canais do modelo assumido.

A maneira sistemática de se avaliar o lado direito de (5.25) é através do conceito apresentado na Seção 5.2.

Como uma aplicação da técnica da medida de distorção média ao problema de avaliação da performance de sistemas de comunicações não codificadas com sinais de resposta parcial periodicamente variantes no tempo é que apresentaremos a seguir os procedimentos a serem seguidos para tal análise.

Primeiramente, se faz necessário identificar os diagramas de estados com erros para cada resposta ao impulso do canal. Diante disso, as matrizes de estado, de condição inicial, de condição de saída e de transição de cada resposta ao impulso do canal são estabelecidas. Sejam Y_i, E_i, S_i e A_i tais matrizes para $i = 1, 2$. Os elementos dessas matrizes são da forma $\alpha_i \cdot D^{\beta_i} z^{\gamma_i}$, onde α_i é uma constante, z um erro de decodificação, β_i distância "Euclidiana" e γ_i

a função distorção.

As equações de estado e de saída quando a política de canal adotada é $\{h_o(t), h_1(t), h_o(t), h_1(t), \ldots\}$ é dada por

$$Y_o = E_o + A_o Y_1 \qquad T_o = S_o \cdot Y_1$$

$$Y_1 = E_1 + A_1 Y_o \qquad T_1 = S_1 \cdot Y_o$$

A função de transferência total é dada por

$$T(z) = \frac{1}{2}\left[T_1(z) + T_2(z)\right]$$

Estamos utilizando $T(z)$ como uma apresentação reduzida de $T(z, D, \gamma_1, \gamma_2)$.

Derivando $T(z)$ acima com respeito à z temos

$$Pb(\gamma_1, \gamma_2) \leq \frac{1}{2} \frac{d}{dz} T(z, D, \gamma_1, \gamma_2)\Big|_{z=1} \qquad (5.26)$$

Como a fonte é binária, a função distorção γ_i iguala a zero se o bit transmitido é igual ao bit decodificado e $\gamma_i = 1$ se eles forem diferentes.

A probabilidade de erro de bit para o sistema de comunicações quando $h_o(t)$ é levado em consideração é dada pelo lado direito de (5.26) fazendo-se $\gamma_1 = 1$ e $\gamma_2 = 0$. Quando $h_1(t)$ é de interesse, então $\gamma_1 = 0$ e $\gamma_2 = 1$. A probabilidade de erro de bit total é a soma dessas duas grandezas, isto é,

$Pb = Pb(1, 0) + Pb(0, 1)$.

Sejam os diagramas de estados com erros como mostrados na Fig. 5.3.1. A partir desses diagramas temos as seguintes equações de estado e saídas.

$$Y_o = \alpha_1 \cdot \frac{z}{2}^{Y_1} + Y_1 \cdot z^{Y_1} \cdot (\alpha_2 + \alpha_3)/2 \qquad T_o = 2\alpha_4 \cdot Y_1$$

$$Y_1 = \beta_1 \cdot \frac{z}{2}^{Y_2} + Y_o \cdot z^{Y_2} \cdot (\beta_2 + \beta_3)/2 \qquad T_1 = 2\beta_4 \cdot Y_o \tag{5.27}$$

e

$$Pb_o = Pb(1, 0) = \frac{1}{2} \frac{d}{dz} T(z, D, Y_1, Y_2)\Big|_{z=1,\ Y_1=1,\ Y_2=0}$$

$$Pb_1 = Pb(0, 1) = \frac{1}{2} \frac{d}{dz} T(z, D, Y_1, Y_2)\Big|_{z=1,\ Y_1=0,\ Y_2=1} \tag{5.28}$$

e $$Pb = Pb_1 + Pb_2$$

Resolvendo-se (5.27) e substituindo em (5.28), após algumas operações algébricas teremos

$$Pb = \frac{1}{4} \cdot \left\{(\alpha_1 \cdot \beta_4 + \alpha_4 \cdot \beta_1) + \frac{1}{4} \cdot (\alpha_4 \cdot \beta_1 + \alpha_1 \beta_4) \cdot (\alpha_2 + \alpha_3) \cdot (\beta_2 + \beta_3) + \right.$$

$$\left. + \alpha_1 \cdot \alpha_4 \cdot (\beta_2 + \beta_3) + \beta_1 \cdot \beta_4 \cdot (\alpha_2 + \alpha_3)\right\} / \left\{1 - \left[(\alpha_2 + \alpha_3) \cdot (\beta_2 + \beta_3) \cdot (1/4)\right]\right\}^2 \tag{5.29}$$

Agora a avaliação final dependerá dos valores de α_i e β_i de (5.29). Para tal iremos assumir que as respostas ao impulso dos canais são dadas por

$$h_o(t) = \begin{cases} \sqrt{(\bar{\alpha}/\bar{\gamma}) \cdot (E/T)} & , \quad \text{para} \quad 0 \leqslant t < T \\ \sqrt{(\bar{\beta}/\bar{\gamma}) \cdot (E/T)} & , \quad \text{para} \quad T \leqslant t < 2T \\ 0 & , \quad \text{para} \quad t \geqslant 2T \end{cases}$$

$$h_1(t) = \begin{cases} \sqrt{(\bar{\delta}/\bar{\gamma}) \cdot (E/T)} & , \quad \text{para} \quad 0 \leqslant t < T \\ \sqrt{(\bar{\eta}/\bar{\gamma}) \cdot (E/T)} & , \quad \text{para} \quad T \leqslant t < 2T \\ 0 & , \quad \text{para} \quad t \geqslant 2T \end{cases}$$

onde $\bar{\alpha} + \bar{\beta} = \bar{\delta} + \bar{\eta} = \bar{\gamma}$. Note que a energia de $h_o(t)$ e $h_1(t)$ vale E.

Com isso, temos que os valores de α_i e β_i são dados por

$$\alpha_1 = \exp\left\{-h_{o,o}/N_o\right\} \qquad \alpha_2 = \exp\left\{-\left(\frac{1}{N_o}\right)(h_{o,o} + 2h_{o,1})\right\}$$

$$\alpha_3 = \exp\left\{-\left(\frac{1}{N_o}\right)(h_{o,o} - 2h_{o,1})\right\} \qquad \alpha_4 = 1$$

$$\beta_1 = \exp\left\{-h_{1,o}/N_o\right\} \qquad \beta_2 = \exp\left\{-\left(\frac{1}{N_o}\right)(h_{1,o} + 2h_{1,1})\right\}$$

$$\beta_3 = \exp\left\{-\left(\frac{1}{N_o}\right)(h_{1,o} - 2h_{1,1})\right\} \qquad \beta_4 = 1$$

Sejam $(\bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\eta})$ números inteiros tais que $\bar{\alpha} + \bar{\beta} = \bar{\delta} + \bar{\eta} = 18$. Note que esses números serão "representativos" da quantidade de energia alocada a cada T seg. de duração da resposta ao impulso dos canais. Para o exemplo em consideração temos que (18, 0, 18, 0) representa o caso de um sistema de comunicações sem interferência entre símbolos, (9, 9, 9, 9) representa o caso de sinais duobinários.

A seguir apresentaremos os resultados da análise feita através dos dados fornecidos nas Tabelas 5.1 e 5.2.

Na Tabela 5.1, comparamos sinais de resposta parcial periodicamente variantes no tempo com período 2 com sinais de resposta parcial invariantes no tempo quando o receptor é o *ótimo*. Como pode ser visto, sinais de resposta parcial periodicamente variantes no tempo apresentam um desempenho melhor do que os invariantes no tempo com ganhos de codificação de até 1.27 db dependendo da combinação de valores de $(\bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\eta})$. Notamos também que existe uma degradação de até 0.44 dB dependendo da combinação periódica selecionada quando comparada com o sistema de comunicações sem interferência. Por outro lado, perdas de até 1.28 dB ocorrem quando comparamos o caso de não interferência entre símbolos com aquela com interferência, porém, invariante no tempo. Deste modo, o ganho líquido é de 0.83 dB quando usamos sinais de resposta parcial periodicamente variantes no tempo ao invés de usar os invariantes no tempo.

Na Tabela 5.2, mostramos a degradação do sistema de comunicações quando ocorre perda de sincronismo. Fica evidente que o melhor desempenho obtido para uma dada combinação de valores de $(\bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\eta})$ passa a ser o pior quando o sincronismo é perdido. Este fato é explicado através do seguinte argumento: os coeficientes da interferência entre símbolos assumem seus menores valores uma vez que as correspondentes métricas acumuladas ao longo dos caminhos na

treliça são menores dado o "descasamento" entre os valores de ramos.

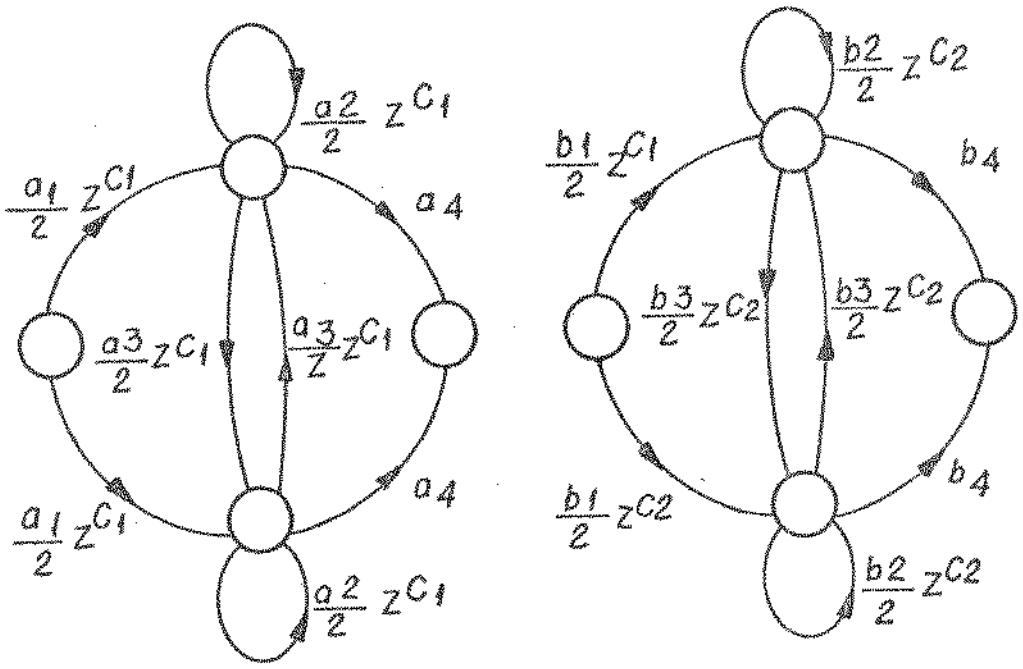


Figura 5.3.1 - Diagrama de estado com erros para L = 2 e M = 2.

TABELA 5.1

Receptor Ótimo

	Pb	10^{-2}	10^{-3}	10^{-4}
	$(\bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\eta})$		RSR (dB)	
	9, 9, 9, 9	7.24	8.8	9.96
	10, 8, 10, 8	7.19	8.75	9.91
	11, 7, 11, 7	7.0	8.6	9.79
	12, 6, 12, 6	6.8	8.45	9.65
	13, 5, 13, 5	6.6	8.3	9.52
	14, 4, 14, 4	6.4	8.15	9.43
	15, 3, 15, 3	6.25	8.05	9.36
	16, 2, 16, 2	6.1	8.0	9.32
	17, 1, 17, 1	6.0	7.95	9.31
Sem Interf.	18, 0, 18, 0	5.96	7.93	9.3

Sinais de Resposta Parcial Invariantes no Tempo

Pb	10^{-2}	10^{-3}	10^{-4}
$(\bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\eta})$		RSR (dB)	
9, 9, 17, 1	6.4	8.18	9.51
10, 8, 17, 1	6.4	8.17	9.5
11, 7, 17, 1	6.35	8.16	9.48
12, 6, 17, 1	6.3	8.15	9.44
13, 5, 17, 1	6.25	8.1	9.4
14, 4, 17, 1	6.2	8.05	9.36
15, 3, 17, 1	6.1	8.0	9.33
16, 2, 17, 1	6.05	7.95	9.32
17, 1, 17, 1	6.0	7.95	9.31
18, 0, 17, 1	5.97	7.95	9.305

Sinais de Resposta Parcial Periodicamente Variantes no Tempo

TABELA 5.2

Receptor Ótimo

Pb	10^{-2}	10^{-3}	10^{-4}	Pb	10^{-2}	10^{-3}	10^{-4}
$(\bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\eta})$	RSR (dB)			$(\bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\eta})$	RSR (dB)		
9, 9, 10, 8	7.21	8.78	9.93	9, 9, 10, 8	8.97	10.96	12.32
← sincronismo perfeito →				← fora de sincronismo →			

5.4 - ANÁLISE DE SISTEMAS DE COMUNICAÇÕES USANDO MODULAÇÃO POR SOBREPOSIÇÃO

O objetivo desta Seção está diretamente relacionado com a aplicação do modelo proposto na Seção 5.2 na avaliação de desempenho de sistemas de comunicações empregando Modulação por Sobreposição e Multi-h.

Especificamente iremos analisar a performance de combinações periódicas não codificadas de modulações Minimum shift keying (MSK) e Binary Phase Shift Keying (BPSK) que pertencem à classe de Técnicas de Modulação de Fase Contínua (CPM). À esta combinação de modulações denominaremos Modulação por Sobreposição.

Modulação por sobreposição é uma técnica proposta onde duas fontes, uma com alta taxa e a outra com baixa taxa, enviam continuamente informação com potência máxima em um mesmo canal de transmissão de tal forma que a ocupação espectral é ligeiramente maior do que aquela da fonte de alta taxa. Dentre as várias combinações de modulações a mais promissora é aquela que emprega o MSK e o

BPSK para altas e baixas taxas respectivamente. Esta combinação foi proposta por Omura e Simon, sem a respectiva análise numérica.

Desta forma, apresentaremos a seguir o modelo do sistema de comunicações para efeito de análise bem como comparações com outras técnicas de modulação digital.

Assumiremos que o modelo do sistema de comunicações seja tal que exista uma periodicidade K e que cada símbolo tenha duração de T segundos. Suponha que a fonte de informação I (alta taxa) transmita durante os K-1 primeiros intervalos de tempo, dentre os K intervalos, e que a fonte de informação II (baixa taxa) transmita somente no K-ésimo intervalo. Iremos assumir também que este padrão se repita ao longo de todo o processo.

O modelo do sistema de comunicações é como mostra a Fig. 5.4.1 com a correspondente treliça.

A saída do sistema de comunicações é dada por

$$x(t, \underline{a}, \underline{b}) = \sqrt{2E/T} \cdot \cos(w_o t + \Theta(t, \underline{a}, \underline{b}))$$

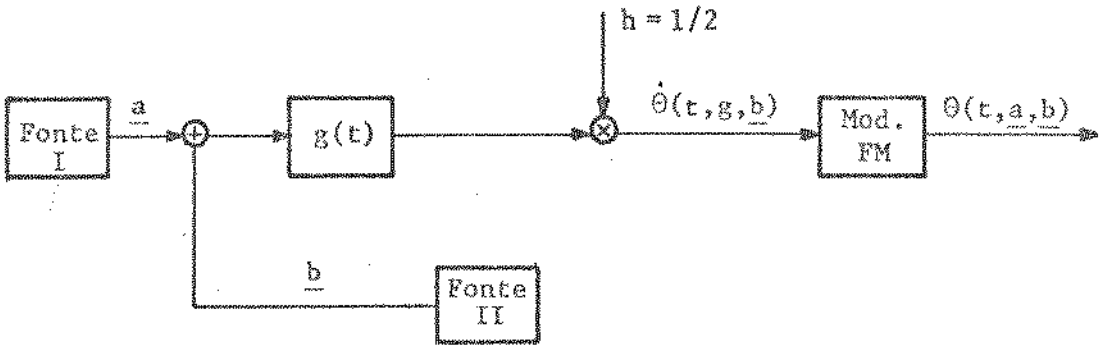
com

$$\underline{a} = (a_o, a_1, a_2, \ldots) \qquad a_i \in \{1, -1\}$$

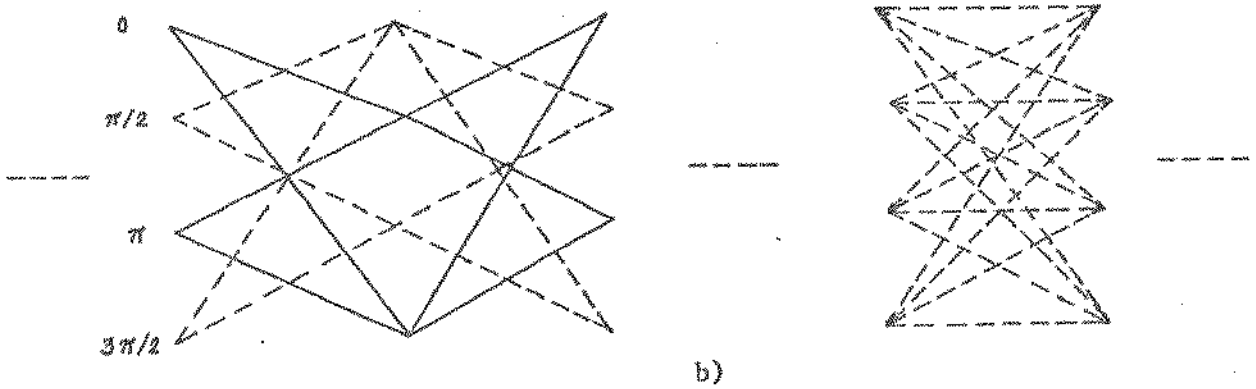
$$\Theta(t, \underline{a}, \underline{b}) = \sum_{n=0}^{j-2} \pi \cdot h \cdot (a_n + b_n) + \int_{(j-1)T}^{t} \pi \cdot h \cdot (a_j + b_j) \cdot q(t - (j-1)T)dt$$

onde

$$b_n = \begin{cases} 0 \text{ ou } 1 & \text{para} \quad n = K \\ 0 & \text{para} \quad n \neq K \end{cases}$$



a)



b)

Figura 5.4.1 - a) Modelo do sistema de comunicações.

b) Diagrama de treliça correspondente.

com K a periodicidade do sistema.

Os estados são dados por

$$s_{n+1} = \left[s_n + \left(\frac{\pi}{2}\right) \cdot (a_n + b_n) \right] \bmod 2\pi \qquad (5.30)$$

Para qualquer sequência de dados provenientes da fonte I e II, o espaço de fase pode ser descrito por uma estrutura de árvore com a característica de fase contínua entre as transições de estados.

Note que esta árvore é binária nos K - 1 intervalos e quaternária no K-ésimo intervalo dada a inclusão de uma das fontes. Esta árvore se converte em uma estrutura de treliça pela operação mod 2π.

Para a simplicidade de análise, suponha que uma base ortogonal apropriada tenha sido escolhida através da aplicação da técnica de ortogonalização de Gram-Schmidt em $x(t, \underline{a}, \underline{b})$. Dessa forma teremos

$$x(t, \underline{a}, \underline{b}) = \sum_{i=0}^{N'} x_i(\underline{a}, \underline{b}) \cdot \phi_i(t)$$

para algum $\{\phi_i(t)\}$ ortogonal.

O canal é o Gaussiano branco com média zero e densidade espectral dos dois lados $N_o/2$.

No receptor teremos:

$$y(t) = x(t, \underline{a}, \underline{b}) + n(t)$$

Dada uma sequência fixa $(\underline{a}, \underline{b})$ de comprimento L e consequentemente dado $x(t, \underline{a}, \underline{b})$, temos que $y(t)$ é um processo Gaussiano com média $x(t, \underline{a}, \underline{b})$ e variância $N_o/2$.

Usando do mesmo conjunto ortogonal de $x(t, \underline{a}, \underline{b})$, isto é, $\{\phi_i(t)\}$, $y(t)$ pode ser representado por

$$\tilde{\underline{y}} = (\tilde{y}_1, \tilde{y}_2, \tilde{y}_3, \ldots, \tilde{y}_L)$$

$$\tilde{\underline{x}}(\underline{a}, \underline{b}) = (\tilde{x}_1(\underline{a}, \underline{b}), \tilde{x}_2(\underline{a}, \underline{b}), \ldots, \tilde{x}_L(\underline{a}, \underline{b}))$$

$$\tilde{\underline{n}} = (\tilde{n}_1, \tilde{n}_2, \ldots, \tilde{n}_L)$$

onde

$$\tilde{y}_k = \tilde{x}_k(\underline{a}, \underline{b}) + \tilde{n}_k$$

com $\tilde{x}_k(\underline{a}, \underline{b})$ e $\tilde{n}_k$ variáveis aleatórias independentes e identicamente distribuídas bi-dimensionais dada por

$$\tilde{\underline{x}}_k(\underline{a}, \underline{b}) = \left[\bar{x}_k(\underline{a}, \underline{b}), \hat{x}_k(\underline{a}, \underline{b})\right]$$

e

$$\tilde{\underline{n}}_k = \left[\bar{n}_k, \hat{n}_k\right]$$

com

$$\bar{x}_k(\underline{a}, \underline{b}) = \sqrt{2E/T} \cdot \cos\left[S_k(a, b)\right]$$

$$\hat{x}_k(\underline{a}, \underline{b}) = \sqrt{2E/T} \cdot \sin\left[S_k(\underline{a}, \underline{b})\right]$$

A função densidade de probabilidade de $\tilde{y}_k$ condicionada em $\tilde{x}_k(\underline{a}, \underline{b})$ é facilmente obtida de $\tilde{y}_k$. Se tomarmos o logaritmo natural desta densidade de probabilidade obteremos uma métrica que é dada por

$$m\left[\tilde{\underline{x}}(\underline{a}, \underline{b}), \tilde{\underline{y}}\right] = \sqrt{2E/T} \cdot \{y_{n,c}(a_n, b_n) \cdot \cos(s_n(\underline{a}, \underline{b})) +$$

$$+ y_{n,s}(a_n, b_n) \cdot \sin(s_n(\underline{a}, \underline{b}))\} = \sum_{n=0}^{L-1} m(x_n, y_n)$$

Como iremos usar o teste de máxima verossimilhança, de modo a determinar a probabilidade de erro de símbolo, teremos que particionar o espaço de sinais em M regiões distintas. Dessa forma o receptor será composto de duas partes

Parte 1 - para todo k fornecerá os valores de $y_{k,c}(a_k, b_k)$ e $y_{k,s}(a_k, b_k)$

Parte 2 - para todo k computará a métrica

$$m\left[\tilde{\underline{x}}(\underline{a}, \underline{b}), \tilde{\underline{y}}\right] = \sum_{n=0}^{L-1} m(x_n, y_n) \quad .$$

que fornecerá uma estimativa da sequência original transmitida.

Uma vez que estabelecemos os elementos básicos para o emprego da técnica da Seção 5.2, passaremos a descrever como obter a função de transferência do diagrama de superestados.

Devido à natureza não linear dos sinais em cada ramo da treliça, cada par de caminhos que deixam um estado e retornam posteriormente a este estado ou terminam em qualquer dos estados restantes da treliça após W interva-

los, com $W \geq K+1$, apresentam diferentes valores para as métricas acumuladas. Isto implica que cada par de caminhos deve ser analisado separadamente.

Suponha que para $t < n$ e $t > n+j$ tenhamos $s_t \neq \tilde{s}_t$ como definido em (5.30). Dessa forma um erro de decodificação ocorrerá se durante os j intervalos, isto é, $n < t < n+j$, o algoritmo de Viterbi escolher um caminho tal que sua métrica acumulada é maior do que a do caminho correto. Assim

$$\sum_{i=n}^{n+j} m(\tilde{x}_i, y_i) \geq \sum_{i=1}^{n+j} m(x_i, y_i)$$

Definiremos $S_k = (\tilde{s}_k, s_k)$ e $U_k = (\tilde{a}_k, a_k)$ como superestado e superentrada no instante $t = k$, respectivamente.

O diagrama de superestados é composto de dois conjuntos a saber W_e e W_{ne} como definidos na Seção 5.2. As matrizes B_i^k, C_i^k e H^k com elementos $h_{i,j}^k$ são respectivamente condição inicial, condição de saída e de transição para $k = 1, 2$. A função de transferência total é dada pelo lema 5.1.

Para a avaliação da performance da modulação por sobreposição basta usarmos diretamente o resultado do lema 5.1. Porém, poderemos simplificar os cálculos se tomarmos a representação de superestado como a representação de diferença de estados. Esta fato permite-nos fazer uso do argumento "os dados de entrada são todos zero", isto é, basta comparar todos os caminhos que deixam o caminho de entradas nulas e retornam a ele. Isto implica que transformamos uma treliça não-linear em uma linear. Portanto, basta determinarmos individualmente as probabilidades de erro de bit e somarmos estes resultados, isto é,

$$PB_\ell = PB(\gamma_1, \gamma_2, \ldots, \gamma_\ell, \ldots, \gamma_j)$$

com $\gamma_\ell = 1$ e $\gamma_i = 0$ para $i = 1, \ldots, j$ e $i \neq \ell$,

e $$Pb = \sum_{\ell=1}^{j} Pb_\ell$$

Para efeito de exemplo numérico iremos estabelecer K = 3. Os diagramas de diferença de estado é mostrado na Fig. 5.4.2.

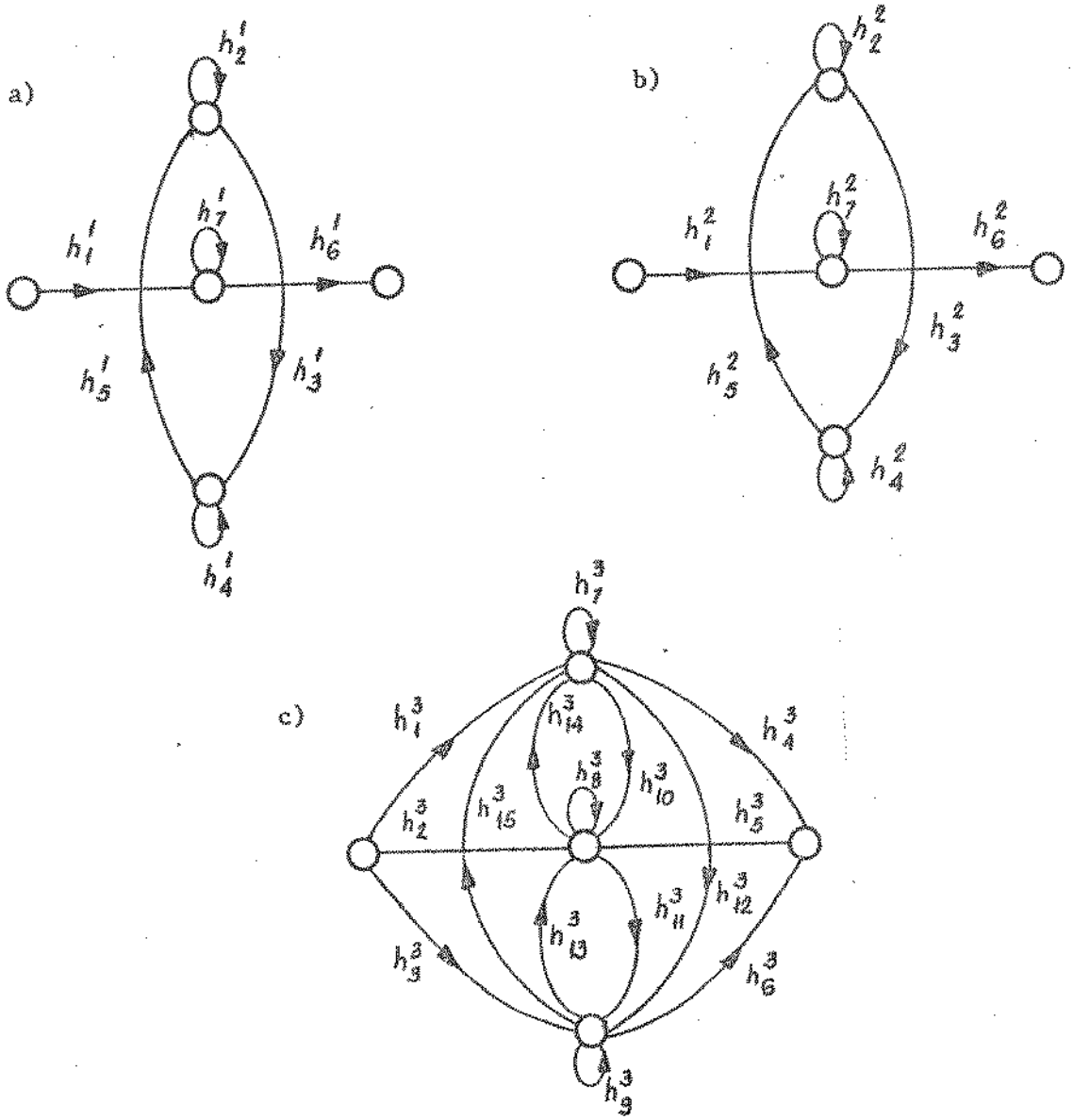


Figura 5.4.2 - Diagramas de diferença de estados.

onde

$$h_1^1 = h_6^1 = D\, z^{\gamma_1} \quad ; \quad h_2^1 = h_4^1 = D \quad ; \quad h_7^1 = D^2$$

$$h_3^1 = h_5^1 = \left(\frac{1}{2}\right) \cdot (D^{.36}\, z^{\gamma_1} + D^{1.64} \cdot z^{\gamma_1})$$

$$h_1^2 = h_6^2 = D\, z^{\gamma_2} \quad ; \quad h_2^2 = h_4^2 = D \quad ; \quad h_7^2 = D^2$$

$$h_3^2 = h_5^2 = \left(\frac{1}{2}\right) \cdot (D^{.36}\, z^{\gamma_2} + D^{1.64}\, z^{\gamma_2})$$

$$h_1^3 = h_3^3 = h_4^3 = h_6^3 = \left(\frac{1}{4}\right) \cdot (3 \cdot D^{.36}\, z^{\gamma_3} + D^{1.21}\, z^{\gamma_3})$$

$$h_{10}^3 = h_{11}^3 = h_{13}^3 = h_{14}^3 = \left(\frac{1}{4}\right) \cdot (3 \cdot D^{1.64}\, z^{\gamma_3} + D^{.79}\, z^{\gamma_3})$$

$$h_{12}^3 = h_{15}^3 = \left(\frac{1}{4}\right) \cdot (2 \cdot D^{1.64}\, z^{\gamma_3} + 2 \cdot D^{.36}\, z^{\gamma_3})$$

$$h_2^3 = h_5^3 = D\, z^{\gamma_3} \quad ; \quad h_7^3 = h_9^3 = D \quad ; \quad h_8^3 = D^2$$

com $D = Q\left[d \cdot \sqrt{\frac{E}{N_o}}\right]$ e $Q(\cdot)$ a função erro, d é a distância Euclidiana normalizada dada por

$$d^2 = \left(\frac{1}{2E}\right) \cdot \int_o^{LT} (x(t, \underline{a}, \underline{b}) - x(t, \underline{a}', \underline{b}'))^2\, dt = \sum_{i=0}^{L-1} d^2\left[(a_i, b_i), (a_i', b_i')\right] \tag{5.32}$$

com $(\underline{a}, \underline{b})$ e $(\underline{a}', \underline{b}')$ sendo quaisquer sequências de comprimento L. Substituindo os valores de $x(t, \underline{a}, \underline{b})$ e $x(t, \underline{a}', \underline{b}')$ em (5.32) temos

$$\tilde{d}^2 = \begin{cases} 1 - \dfrac{(\operatorname{sen} \Delta\theta_{i+1} - \operatorname{sen} \Delta\theta_i)}{\Delta\theta_{i+1} - \Delta\theta_i} & , \quad \Delta\theta_i \neq \Delta\theta_{i+1} \\ 1 - \cos \Delta\theta_i & , \quad \Delta\theta_i = \Delta\theta_{i+1} \end{cases}$$

onde $\tilde{d}^2 = d^2\left[(a_i, b_i), (a_i', b_i')\right]$ e $\Delta\theta_i = s_i - \tilde{s}_i$. De (5.30) temos que $\Delta\theta_i$ é dado por

$$\Delta\theta_{n+1} = s_{n+1} - \tilde{s}_{n+1} = \sum_{i=0}^{n} \eta_i \cdot \left(\frac{\pi}{2}\right)$$

onde η_i e suas respectivas probabilidades de transição, representadas entre colchetes, são dadas por

$$\eta_i, \left[q(\eta_i)\right] = \begin{cases} 0 \quad [1] & , \quad a_i = a_i' \ , \quad b_i = b_i' \\ 1 \quad [3/4] & , \quad \begin{matrix} a_i = a_i' \ , & b_i \neq b_i' \\ a_i \neq a_i' \ , & b_i = b_i' \end{matrix} \\ 2 \quad [2/4] & , \quad a_i \neq a_i' \ , \quad b_i = b_i' \\ 3 \quad [1/4] & , \quad a_i \neq a_i' \ , \quad b_i \neq b_i' \end{cases}$$

Em geral $q(\eta_i)$ é dada por

$$q(\eta_i) = 1 - \frac{|\eta_i|}{M \cdot c}$$

onde c iguala o intervalo entre os valores de η_i. No caso em consideração c = 1.

Uma vez que dispomos de todos os valores para avaliar o limitante superior da probabilidade de erro de bit, passaremos às comparações. A Fig. 5.4.3 mostra Pb versus $\frac{E_b}{N_o}$ para modulação por sobreposição com K = 3, 4, MSK, BPSK, QPSK e 2-h dada por {2/4, 3/4} avaliado segundo a) e b) da Fig. 5.4.2.

Com relação às comparações temos que as técnicas de modulações digitais convencionais, isto é, M-PSK, M-FSK, QAM e etc. apresentam discontinuidade de fases abruptas. Estas discontinuidades resultam numa maior ocupação espectral quando comparadas com técnicas de modulação digital pertencentes à classe CPM.

Como pode ser observado da Fig. 5.4.3, para valores razoáveis da periodicidade K a modulação por sobreposição está muito próxima do MSK uma vez que para tais valores o termo predominante é da forma $\left(\frac{1}{2}\right) \exp\left(-\frac{E_b}{N_o}\right)$. Dessa forma temos que a modulação por sobreposição propicia um desempenho bom com uma faixa de ocupação espectral ligeiramente maior do que aquela do MSK, porém, transmitindo duas fontes com alta e baixa taxas.

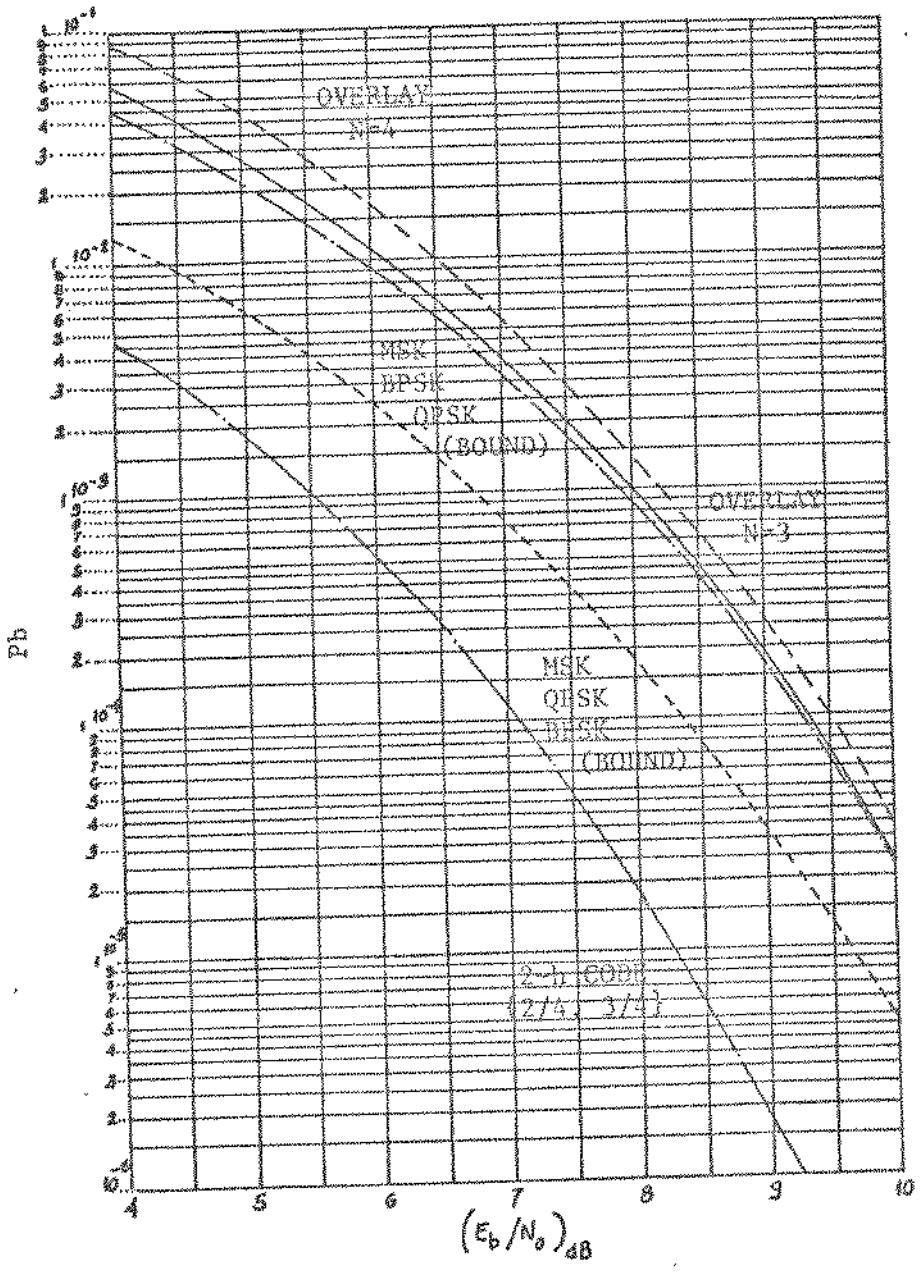


Figura 5.4.3 - Pb versus E_b/N_o.

5.5 - ANÁLISE DE DESEMPENHO DE ESQUEMAS COMBINADOS POLLING/CÓDIGOS CONVOLUCIONAIS PARA CANAIS DE COMUNICAÇÕES DE MÚLTIPLO ACESSO

Nesta Seção apresentaremos a análise de desempenho de sistemas de comunicações combinando técnicas de múltiplo acesso denominada "Polling" juntamente com códigos convolucionais. As disciplinas do "Polling" denominadas "gated" e exaustiva serão consideradas juntamente com códigos convolucionais com moderados valores de memória e taxas específicas. As medidas de desempenho consistem do atraso médio e da probabilidade de erro de bit como uma função da faixa e potência.

Iremos considerar um sistema de múltiplo acesso cujo objetivo é o de partilhar os recursos entre elementos geograficamente distribuídos. Em geral, técnicas de codificação são incorporadas à um sistema de comunicações de dados de modo a melhorar a confiabilidade das mensagens ao passarem por um meio com as interferências e ruído.

Em termos de confiabilidade, a medida de desempenho é a probabilidade de erro de bit, Pb, como uma função da relação sinal-ruído (E_b/N_o). Por outro lado, várias curvas de desempenho do tipo Pb x E_b/N_o, tem sido obtidas para vários esquemas de codificação. Entretanto, tais derivações tem assumido, em geral, que o fluxo de dados (mensagens) que chegam ao canal de comunicações é *constante*. Sob condições reais, a geração de dados nas várias estações (usuários) são fornecidas por várias fontes de informação e governadas por diferentes mecanismos aleatórios. Como consequência, comprimentos das mensagens e os intervalos entre as chegadas são representadas por variáveis aleatórias. Desse modo, formação de filas surgirão, isto é, as mensagens sofrerão atrasos nas estações da rede.

O tempo médio de atraso das mensagens $\bar{D}$ fornece a média estatística do tempo decorrido a partir do instante da requisição para a transmissão da mensagem até o instante que ela é recebida com sucesso pelo destinatário.

Note que além da formação de filas, existe um outro fato que degradará o desempenho do sistema. Dado que as fontes estão geograficamente distribuídas e são independentemente geradas alguma política deve ser adotada de modo a organizar o acesso ao canal. Isto implica que deverá existir uma coordenação entre os usuários. Consequentemente, atraso das mensagens nas filas será afetado pelo modo de múltiplo acesso adotado.

Compartilhamento de um canal de múltiplo acesso, canal de radiodifusão, pelas estações é determinado e supervisionado através do procedimento de controle de acesso. Governado pelo protocolo de acesso, cada estação discriminará o tempo e/ou a frequência apropriadas para efetuar a transmissão. A medida mais comum de desempenho de um procedimento de controle de acesso consiste do tempo médio de resposta das mensagens na rede, $\bar{D}$, como função do tráfico total efetivamente transportado (throughput).

Para a descrição apropriada do desempenho combinado de um sistema de comunicações de múltiplo acesso governado conjuntamente pelo controle de acesso/técnicas de codificação, devemos combinar as curvas de Pb x E_b/N_o com as curvas de atraso versus "throughput".

Trabalhos anteriores apresentaram resultados somente para os casos de TDMA e FDMA com esquemas de modulações PSK e FSK. Nesta Seção iremos apresentar os resultados para "Polling" integrado com códigos convolucionais. O propósito desta integração advém do fato de que ambas as técnicas são utilizadas de forma independente em várias aplicações de comunicações de dados com multi-usuários de tal forma que os projetistas de redes, analistas ou mesmo usuários de-

terminem a quantidade total de informação que possa ser transmitida através do canal de uma forma confiável e regular.

Os outros parâmetros importantes que serão considerados nesta análise são faixa do canal, R bits/seg., e potência, C/N_o.

Consideraremos o sistema de múltiplo acesso como servindo uma população de M usuários (estações) enumeradas de 1 a M. Para simplicidade matemática, assumiremos que todas as estações possuem as mesmas características. Deve ser ressaltado que esta hipótese de forma alguma inviabilizará a análise a ser feita.

Iremos considerar que a chegada das mensagens em cada estação segue o processo de Poisson com taxa λ. Então,

$\lambda = \lambda_i$ (mess./seg.) = taxa de chegada em cada terminal

$\Lambda = M \cdot \lambda$ = taxa total de chegada no sistema

Cada mensagem é composta de um número de pacotes. Denotaremos por B_n o número de pacotes contido na n-ésima mensagem que chega ao sistema. As variáveis aleatórias $\{B_n;\ n = 1, 2, \ldots\}$ são independentes e identicamente distribuídas governadas por uma distribuição geral com momentos $E(B)$ e $E(B^2)$.

Cada pacote é composto de um cabeçalho com N_h bits e um campo de dados com N_d bits. Deste modo o número total de bits num pacote é $N_h + N_d$.

A taxa de transmissão através do canal é R bits/seg. Assim, o tempo necessário para transmitir uma mensagem é uma variável aleatória χ dada por

$$\chi = (N_h + N_d) \cdot B/R$$

e cujos dois primeiros momentos são

$$E(\chi) = (N_h + N_d) \cdot E(B)/R$$

$$E(\chi^2) = \left[\frac{(N_h + N_d)}{R}\right]^2 \cdot E(B^2)$$

O direito de acesso ao canal é controlado e supervisionado pelo protocolo do "Polling". Sob este protocolo, que pode ser implementado na forma centralizada ou distribuída, cada estação é interrogada, em uma forma cíclica, se tem alguma mensagem a ser transmitida. Se a resposta é positiva, a faixa é totalmente alocada a esta estação, que transmitirá na taxa de R bits/seg. Após a utilização do canal pela estação i (ou caso haja uma resposta negativa à interrogação), o mesmo procedimento é repetido para a estação (i + 1) mod M. Uma vez que o acesso é alcançado, duas disciplinas podem ser consideradas: "gated" e a exaustiva. Na disciplina "gated" somente as mensagens que chegaram até o instante de interrogação são transmitidas neste ciclo. Na disciplina exaustiva, as mensagens que chegam durante a transmissão são também transmitidas no presente ciclo.

Cada mensagem sofrerá um atraso total devido ao tempo de espera mais o tempo de transmissão. Em regime permanente, o atraso médio é representado por $\bar{D}$.

Até aqui, apresentamos o protocolo de múltiplo acesso. Iremos a seguir apresentar o esquema de codificação.

Em cada estação a técnica de codificação do tipo convolucional é aplicada às mensagens. Cada sequência de b bits que alimenta o codificador tem

como resposta n bits. Dessa forma estamos utilizando codificadores convolucionais com taxa $R_c = b/n$, b e n inteiros. Um outro parâmetro importante é o número de memórias, ν, de tal forma que um total de $b \cdot \nu$ bits influenciam a saída do codificador.

Assumiremos que cada bit tenha a duração de T_b seg. Isto é equivalente a dizer que a taxa de geração dos bits é $R_b = 1/T_b$ bps. Relembrando que o tempo necessário para transmitir um bit através do canal é $T = 1/R$, obtemos para o codificador em questão

$$T_b = R_c \cdot T = R_c/R$$

Então, assumindo que o conteúdo de energia do bit é E_b, que as características de atenuação do canal e ganho da antena são tais que a relação sinal-ruído total, seja C/N_o, temos que

$$\frac{C}{N_o} = (E_b/T_b)/N_o \longrightarrow \frac{E_b}{N_o} = (C/N_o) \cdot R_c/R$$

Desta forma a medida de desempenho do sistema sob tais considerações fica como uma função de C/N_o, ou equivalentemente, de E_b/N_o.

Agora passaremos a análise de desempenho propriamente dita. Então sob o modelo adotado, temos para a média estatística do atraso da mensagem para a disciplina "gated" fica

$$\bar{D}_q = \left(\frac{1}{2}\right) \cdot \frac{M \lambda E(\chi^2)}{1 - \rho} + \left[1 + \lambda E(\chi)\right] \cdot M \cdot r \quad (1 - p) + E(\chi)$$

onde

$$\rho = M \lambda E(\chi) \leq 1$$

e r é o tempo necessário para passagem de controle entre as estações. Esta hipótese somente simplifica a exposição e a análise a ser desenvolvida. Entretanto, para o caso geral r deve ser considerada como uma variável aleatória. Pode-se mostrar que o atraso médio sob a disciplina exaustiva é dado por

$$\bar{D}_e = \bar{D}_q - \frac{\rho \cdot r}{1 - \rho}$$

Como a relação acima leva em conta o atraso médio sob a disciplina "gated", então é suficiente efetuarmos a análise sob a disciplina "gated" para obtenção a posteriori do atraso médio da disciplina exaustiva.

Defina

$$\Lambda_b = M \cdot \lambda \cdot (N_h + N_d) \cdot E(B) = \rho \cdot R$$

como o tráfico médio total (taxa de bits de chegada) transportado pela rede (em bits por seg.). Notamos que

$$\bar{D}_q = d\left[R, \Lambda_b, E(B), E(B^2), M\right]$$

onde $d[\cdot]$ é uma função monotonicamente decrescente em R e monotonicamente crescente em Λ_b, e consequentemente em $\rho = \Lambda_b/R$ para $\rho < 1$. Isto implica que a inversa de $d[\cdot]$ com respeito a Λ_b existe. Seja $d^{-1}[\cdot]$ a inversa, então

$$\Lambda_b = d^{-1}[R, \bar{D}_q, E(B), E(B^2), M]$$

é a taxa de bits da rede em termos de $\bar{D}_q$. Além disso, Λ_b é monotonicamente crescente com $\bar{D}_q$ e Λ_b tende a R quando $\bar{D}_q \to \infty$.

Com relação aos códigos convolucionais, limitante superior da medida de distorção média pode ser facilmente obtidos da Seção 5.2 quando conhecemos a matriz geradora, G, do código para um dado número de memória, ν, taxa $R_c = \frac{b}{n}$ e canal. Tabelas com novos códigos convolucionais foram apresentadas no Capítulo 2. Entretanto, limitaremos a análise somente com o emprego de alguns códigos.

Dessa forma sob as hipóteses do modelo adotado juntamente com a modulação BPSK e assumindo que o ruído é aditivo Gaussiano branco pode-se mostrar que para : 1) $\nu = 3$, $R_c = 3/4$; 2) $\nu = 4$, $R_c = 2/3$; 3) $\nu = 1$, $R_c = 1/2$ o limitante superior de $\hat{d}$ quando a função característica é usada é dada por:

1) $$Pb \leq \left(\frac{1}{2}\right) \cdot \{1511\ D^8 + 56663\ D^{10} + 2119829\ D^{12} + \ldots.\}$$

2) $$Pb \leq \left(\frac{1}{2}\right) \cdot \{265\ D^8 + 5495\ D^{10} + 110834\ D^{12} + \ldots.\}$$

3) $$Pb \leq \left(\frac{1}{2}\right) \cdot \{36\ D^{10} + 211\ D^{12} + 1404\ D^{14} + 11633\ D^{16} + \ldots.\}$$

onde

$$D = Q\left(\sqrt{\frac{2\ E_b}{N_o}}\right)$$

e $Q(\cdot)$ sendo a função erro.

Observamos que $Pb = p\left[(C/N_o) \cdot R_c/R\right]$, onde $p(\cdot)$ é monotonicamente decrescente com C/N_o. Então, sua inversa $p^{-1}(\cdot)$ existe e é dada por:

$$\frac{C}{N_o} \cdot \frac{R_c}{R} = p^{-1}(Pb)$$

Note que a conecção entre o desempenho de $\bar{D}_q$ x ρ (expresso por $d(\cdot)$) e Pb x $\frac{C}{N_o}$ (expresso por $p(\cdot)$) é dado pela taxa do canal R. No que se segue, ilustraremos o desempenho combinado e as transações associadas com essas funções desempenho.

De modo a apresentarmos resultados numéricos para o desempenho combinado de códigos convolucionais e "Polling", consideraremos um sistema com os seguintes parâmetros : $M = 50$; $r = 10^{-6}$ seg.; número de pacotes por mensagem tem distribuição geométrica com média $E(B) = 2$ (implicando que $E(B^2) = 5$); $N_d = 1000$ bits; $N_h = 48$ bits.

Primeiramente, note que $d(\cdot)$ é função de Pb. Deste modo as curvas de $\bar{D}_q$ x ρ (parametrizadas em Pb) são dadas por

$$\bar{D}_q = d\left[R, \Lambda_b, E(B), E(B^2), M\right]$$

com

$$R = \left(\frac{C}{N_o}\right) \cdot \frac{R_c}{p^{-1}(Pb)} > \Lambda_b$$

Então, avaliação e computo do atraso versus "throughput" sob um determinado va-

lor de R, as relações acima integram os elementos da combinação.

A Fig. 5.5.1 mostra a variação de $\bar{D}_q$ como uma função da throughput média da rede em bps. Este último parâmetro é a taxa efetiva de transmissão de bits e pode ser escrita como $R - M\ \lambda\ N_d\ E(B)$. Note que $M\ \lambda\ N_h\ E(B)$ é a quantidade da capacidade de canal desperdiçada pela transmissão dos bits do cabeçalho.

A variação de Pb x $\frac{C}{N_o}$ é determinada pela função $p(\cdot)$ tal que

$$Pb = p\left[\left(\frac{C}{N_o}\right) \cdot \frac{R_c}{R}\right]$$

Assim, para o máximo valor do atraso médio, $\bar{D}_q$, as curvas de desempenho do atraso parametrizado pelas curvas de Pb x $\frac{C}{N_o}$ são obtidas com R dado por

$$R = \frac{\Lambda_b}{\rho}$$

onde Λ_b é que fornecerá o valor de $\bar{D}_q$. Isto é equivalente a escolher o valor de R para um dado valor de $\bar{D}_q$. Valores pequenos de R resultarão em atrasos médios maiores que o de $\bar{D}_q$. Note que, quando $\bar{D}_q \to \infty$, $R \to \Lambda_b$ e consequentemente, $Pb \to Pb\ (min)$;

$$Pb(min) = p\left[\left(\frac{C}{N_o}\right) \cdot \frac{R_c}{\Lambda_b}\right]$$

Deste modo Pb(min) dá o desempenho do esquema de codificação quando

nenhuma restrição de atraso é imposta.

Na Fig. 5.5.2, ilustramos o comportamento de Pb x $\left(\frac{C}{N_o}\right) \cdot \Lambda_b^{-1}$. Os parâmetros são os mesmos que da Fig. 5.5.1. As curvas apresentadas são para vários valores da média estatística do tempo de espera das mensagens, $\bar{W}_q = \bar{D}_q - E(\chi)$. Observamos que se nenhuma restrição de atraso é imposta, tal que $\bar{W}_q = \infty$, as curvas convencionais de Pb x $\frac{C}{N_o}$ são obtidas. Por outro lado, se $\bar{W}_q$ = 3 m seg., observamos que há degradação em Pb para um dado $\frac{C}{N_o}$. As trocas entre $\bar{W}_q$ e Pb tornam-se evidentes neste caso.

Finalmente, ilustramos as possíveis trocas entre as duas principais medidas de desempenho, $\bar{D}_q$ e Pb. Para valores de $\frac{C}{N_o}$, R_c e Pb dados assumiremos que R possa assumir seu valor máximo. Assim, usando este valor de R, é possível a obtenção da relação entre o atraso e Pb que é dada por:

$$\bar{D}_q = d\left[\left(\frac{C}{N_o}\right) \cdot \frac{R_c}{p^{-1}(Pb)} \;,\; \Lambda_b \;,\; E(B) \;,\; E(B^2) \;,\; M\right]$$

Note que se escolhermos o maior valor de R, garantiremos o menor valor de $\bar{D}_q$.

Para obtenção de valores finitos de $\bar{D}_q$, precisamos garantir que $\rho = \Lambda_b/R < 1$. Então

$$Pb > \hat{Pb} = p\left[\left(\frac{C}{N_o}\right) \cdot \frac{R_c}{\Lambda_b}\right]$$

Na Fig. 5.5.3, mostramos o comportamento de $\bar{W}_q$ x Pb para $\frac{C}{N_o}$ = 50 dB.

Os parâmetros aqui utilizados são os mesmos das figuras anteriores. Observamos que para um código convolucional com $\nu = 6$ e $R_c = 1/2$ o melhor desempenho é obtido, porém, necessita de maior faixa.

Como, em geral, códigos convolucionais com altas taxas alcançam as menores distâncias mínimas do que os códigos com baixas taxas e que o fator de expansão da faixa é dado pelo inverso de R_c, então concluimos que se o sistema requer alta confiabilidade, baixas taxas tem que ser utilizadas tendo em conta o fator de expansão de faixa. Se a faixa tem que ser conservada então códigos convolucionais com altas taxas é a alternativa. Se preservação de faixa é necessária, porém, com ganhos em potência então códigos de treliça invariantes no tempo e periodicamente variantes no tempo são alternativas fantásticas.

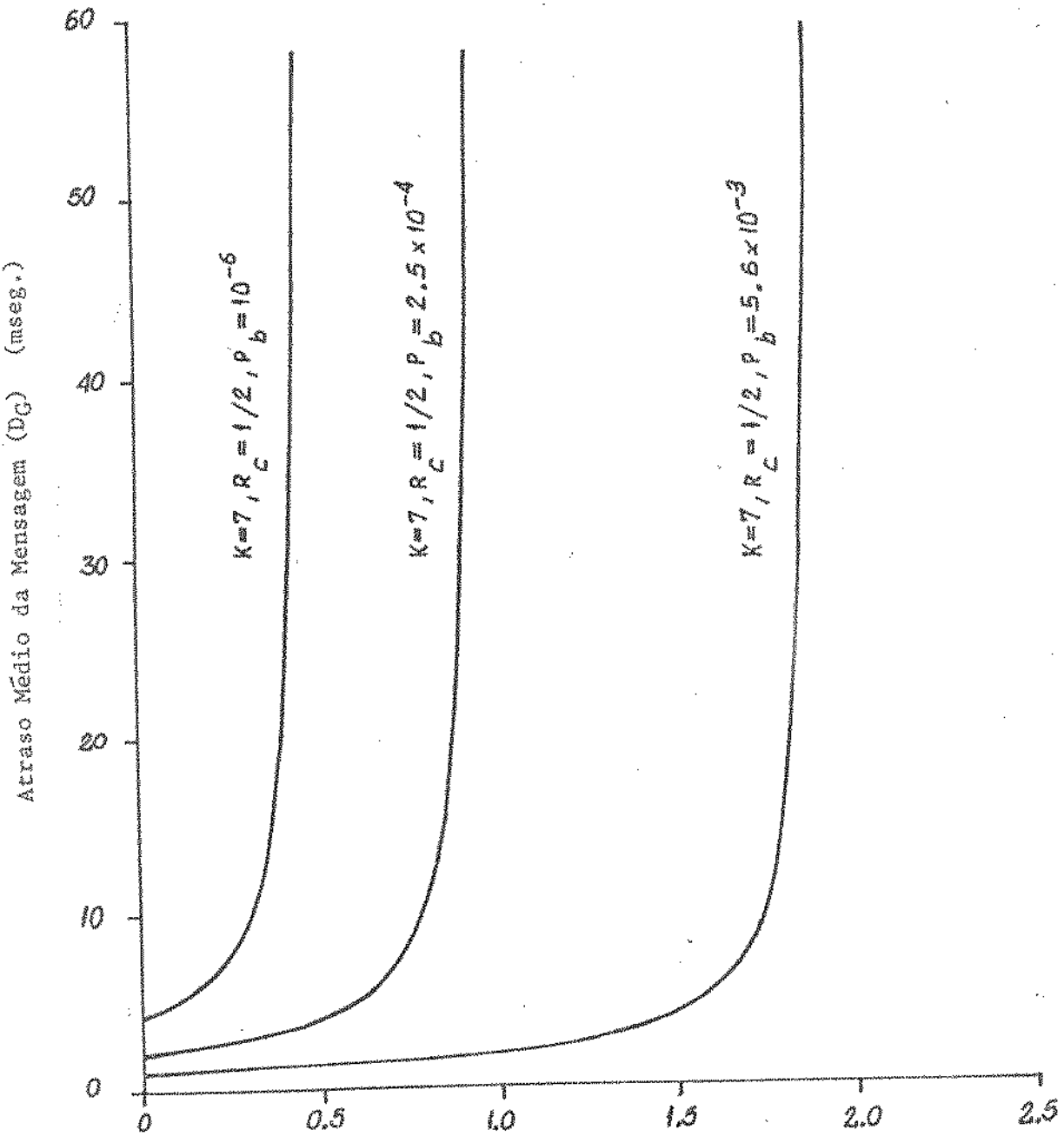


Figura 5.5.1 - Atraso da mensagem x "throughput" para C/N_o = 60 dB com códigos convolucionais ν, R_c e Pb especificados.

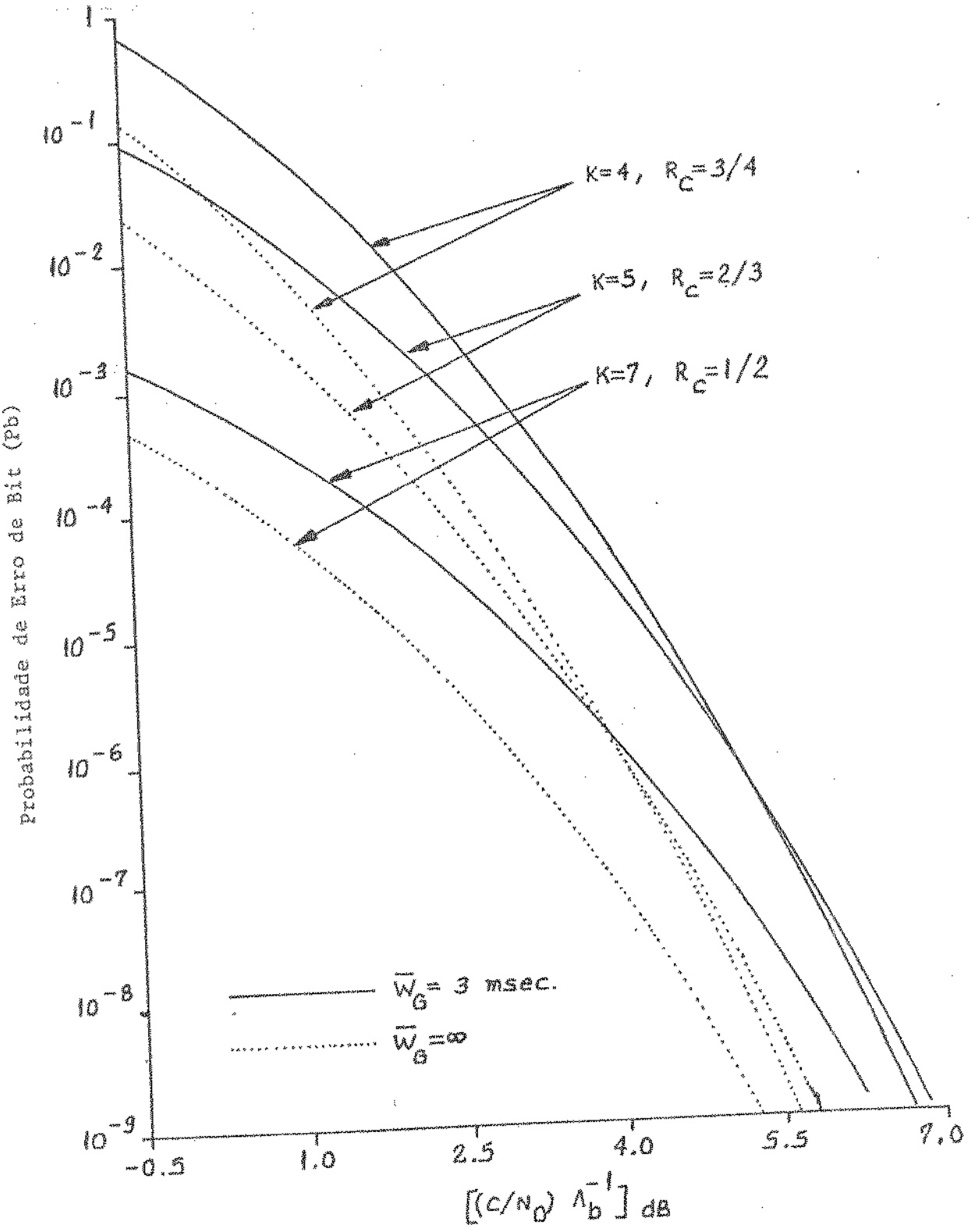


Figura 5.5.2 - Probabilidade de erro de bit x $(C/N_0)\cdot\Lambda_b^{-1}$ para códigos convolucionais com ν, R_C especificados; com e sem restrição de espera.

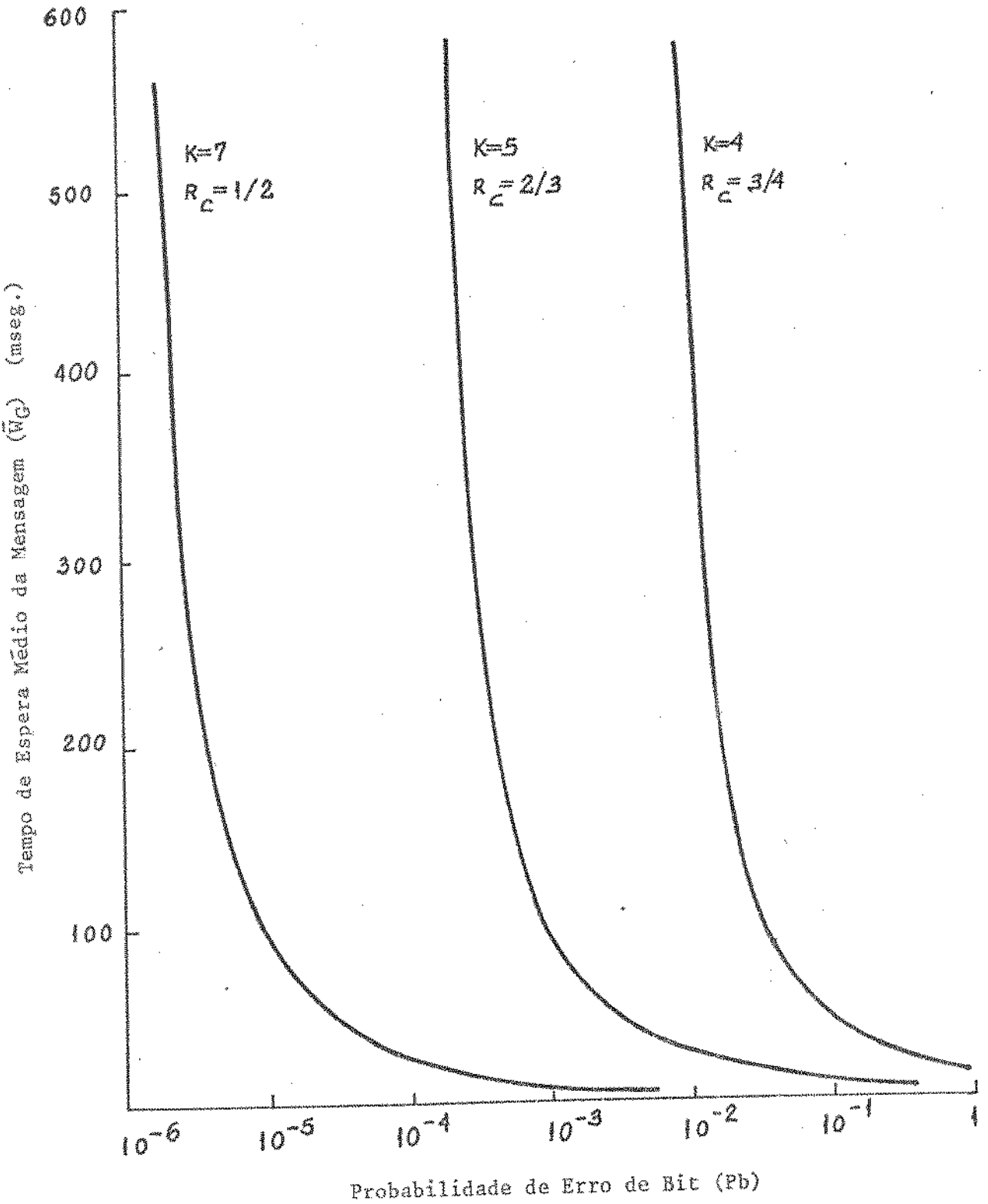


Figura 5.5.3 - Tempo de espera da mensagem x Pb com combinação da disciplina "gated" e códigos convolucionais com ν, R_c especificados e (C/N_o) = 50 dB.

Referências

"Performance Analysis of Periodically Time Varying Partial Response Signaling with Maximum Likelihood and Integrate & Dump Receivers," *IEEE International Symposium on Information Theory*, Ann Arbor, Michigan, USA, Outubro 5-9, 1986.

"An Upper Bound on the Average Distortion Function Via Dynamic Programming," *First International Symposium on Operation Research*, Santa Maria, R.S., Brasil, Junho 6-8, 1986.

"Combined Performance Analysis of Joint Polling/Convolutional Coding Schemes in Multiple Access Communication Channels," *IEEE INFOCOM 86*, Flórida, USA. (com L.F. de Moraes). Abril 10-14, 1986.

"Performance Analysis of PAM Communication Systems with Periodically Time Varying Partial Response Signals," *2nd IASTED International Conference on Telecommunication and Control*, Rio de Janeiro, Brasil, Dezembro 10-13, 1985.

"Performance Analysis of Overlay Modulation and Comparison with other Modulation Techniques," *International Communications and Energy Conference*, Montreal, Canadá, Outubro 2-4, 1984.

"Dynamic Transfer Function Technique Applied to Periodically Time Varying Convolutional Codes" *IEEE International Symposium on Information Theory*, Canadá, Saint Jovite, Quebec, Canadá, Setembro 1983.

"Real Nonlinear Trellis Codes" *Primeiro Simpósio Brasileiro de Telecomunicações*, Rio de Janeiro, Setembro, 1983.

"Bit Error Probabilities of Binary Linear Trellis Codes" *Primeiro Simpósio Brasileiro de Telecomunicações*, Rio de Janeiro, 1983.

"Performance Evaluation when Instantaneous Phase Jumps are Allowed in the Trellis" Impromptum Section, *IEEE International Symposium on Information Theory*, Les Arcs, France, 1982.

"Estudos Comparativos de Sistemas MAPSK (1, 3) e (1, 7) tanto Teórico como Prático," XXIX Reunião Anual da SBPC, 1977.

"Deteção e Sincronização de Portadora APK," XXVII Reunião Anual da SBPC, 1976.

"Estudos de Otimização de Conjuntos de Sinais APK," XXVIII Reunião Anual da SBPC, 1976.

"Cancelamento de Eco : Considerações Preliminares" Relatório Técnico No. 26, DEE-FEC-UNICAMP, Maio, 1986.

"Cancelamento de Eco : Algoritmo de Máxima Verossimilhança" Relatório Técnico No. 28, DEE-FEC-UNICAMP, Junho, 1986.

"Cancelamento de Eco: Algoritmo de Mínimo Quadrado para Estruturas Lattice" Relatório Técnico No. 27, DEE-FEC-UNICAMP, Julho, 1986.

"Trunk Link Protocol Models" Relatório Técnico do Projeto de Pesquisa entre Hewlett-Packard e Technology Transfer Institute, Los Angeles, California, Maio, 1984, (com J. Silvester).

"On Channel Modelling" Relatório Técnico do Projeto Hewlwtt-Packard e Technology Transfer Institute, Los Angeles, California, Junho, 1984.

"General Analysis of Selective Repeat Error Control Schemes," Relatório Técnico do Projeto de Pesquisa entre Hewlett-Packard e Technology Transfer Institute, Los Angeles, California, Agosto, 1984.

"Preliminary Studies for Improving Performance of the Quotron System Network," Relatório Técnico Quotron Systems Inc., Los Angeles, California, Abril, 1984.

"Improving Performance of the QSN Network by Using Alphabet Redundant Codes," Relatório Técnico Quotron Systems Inc., Los Angeles, California, Junho, 1984.

"Transmissão de Sinais PCM Via Rádio," Relatório Técnico No. 14, Publicação Interna FEC/UNICAMP, 1977.

"Estudos de Modulação e Demodulação de Sinais na Transmissão Via Rádio," Publicação Interna FEC/UNICAMP, No. 07/77, 1977.

"General Dynamic Transfer Function Technique for Periodically Time Varying Convolutional Codes," *IEEE Transactions on Information Theory*, submetido para publicação.

"Periodically Time Varying Partial Response Signalling," *IEEE Transactions on Communications*, submetido para publicação.

"Periodically Time Varying Ungerboeck Codes better than any other Time Invariant Ones," *IEEE Transactions on Information Theory*, submetido para publicação.

"Performance Analysis of New Digital Modulation Schemes," *IEEE Transaction on Communications*, submetido para publicação.

"Some Good Time Varying TCM Codes", *International Symposium on Information and Coding Theory*, Campinas Julho 27-Agosto 1, 1987. (submetido) (colaboradora : Keiko V.O. Fonseca)

"Unequal Error Protection of TCM Codes", *IEEE Transactions on Information Theory*, a ser submetido para publicação.

CAPÍTULO 6

CONCLUSÕES

Os capítulos anteriores resumem de forma suscinta as contribuições segundo a linha de pesquisa desenvolvida pelo autor.

Esta linha de pesquisa pode ser descrita como uma divisão em três áreas:

1 - Otimização Combinatorial

2 - Criptografia

3 - Códigos de Treliça Variantes no Tempo

Não foram incluídas neste trabalho, por razões de espaço, os resultados recentes dos novos tópicos de pesquisa em desenvolvimento que apresentam resultados bastante promissores tanto no aspecto teórico como aplicado.

Dentre esses novos tópicos de pesquisa citamos:

a) Códigos de treliça variantes no tempo para uso em Múltiplo Acesso;

b) Estabilidade e robustez de códigos convolucionais com aplicações em criptografia;

c) Generalizações de cripto-sistemas de chaves Pública variantes no tempo;

d) Processos adaptativos usando o algoritmo de Viterbi para cancelamento de eco e/ou interferência intersimbólica;

e) Processos adaptativos para codificação de fontes usando códigos de treliça variantes no tempo;

f) Algoritmos para a solução do "knapsack" combinatorial.

etc. como exemplos.