

Capítulo 8

Redes de Hopfield

O texto apresentado neste capítulo segue basicamente Haykin em [1], sendo uma tradução e interpretação livres e resumidas dos conteúdos lá apresentados.

Uma Rede de Hopfield consiste de um conjunto de neurônios e um correspondente conjunto de unidades de atraso, formando um sistema de realimentação múltiplo.

A Figura 8.1 apresenta a arquitetura de uma Rede de Hopfield formada de $N = 4$ neurônios. Observe que o número de laços de realimentação é igual ao número de neurônios. Basicamente, a saída de cada neurônio é realimentada através de um elemento de atraso, para cada um dos outros neurônios da rede, sem haver auto-realimentação.

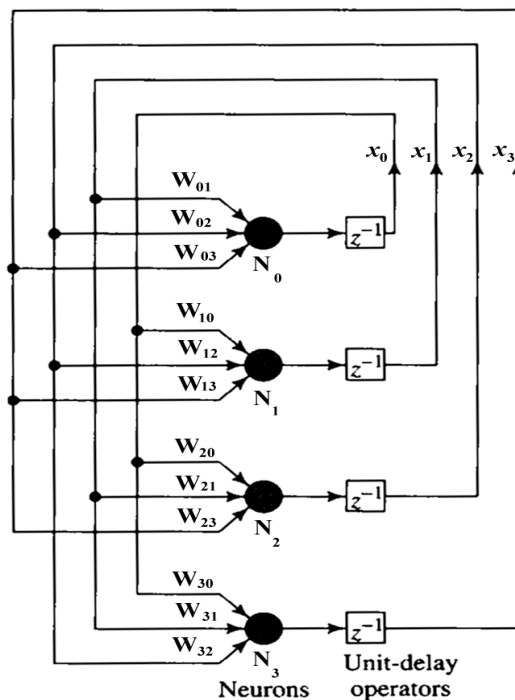


Figura 8.1: Arquitetura de uma Rede de Hopfield formada de $N = 4$ neurônios.

Ao estudar a dinâmica da Rede de Hopfield utiliza-se o modelo neurodinâmico aditivo, que é o modelo de neurônio sem ruído mostrado na Figura 8.2.

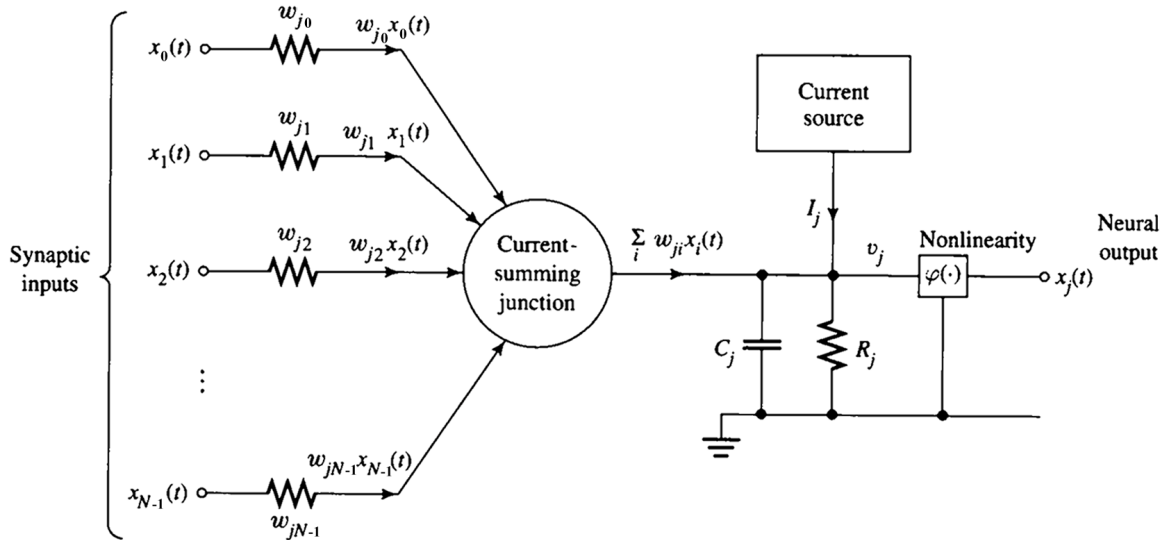


Figura 8.2: Modelo aditivo de um neurônio.

Em termos físicos, os pesos sinápticos $w_{j0}, w_{j1}, \dots, w_{jN-1}$ (mostrados na Figura 8.2) representam condutâncias e as respectivas entradas $x_0(t), x_1(t), \dots, x_{N-1}(t)$ representam potenciais (N é o número de entradas). Estas entradas são aplicadas a um nó somador de corrente que possui as seguintes características:

- Baixa resistência de entrada.
- Ganho de corrente unitário.
- Alta resistência de saída.

O fluxo total de corrente entrando no nó v_j na Figura 8.2 é:

$$\sum_{i=0}^{N-1} w_{ji} x_i(t) + I_j$$

onde o primeiro termo da soma é devido aos estímulos $x_0(t), x_1(t), \dots, x_{N-1}(t)$ atuando sobre os pesos sinápticos (condutâncias) $w_{j0}, w_{j1}, \dots, w_{jN-1}$, respectivamente, e o segundo termo é devido à fonte de corrente I_j que representa um *bias* aplicado externamente.

Seja $v_j(t)$ a função no tempo que descreve o campo local induzido no nó v_j . O fluxo total de corrente saindo do nó v_j é:

$$\frac{v_j(t)}{R_j} + C_j \frac{dv_j(t)}{dt}$$

onde o primeiro termo é devido à resistência R_j e o segundo termo é devido à capacitância C_j .

Pela lei de corrente de Kirchoff, sabemos que o fluxo de corrente total que entra em qualquer nó de um circuito elétrico é zero. Aplicando a lei de corrente de Kirchoff ao nó v_j na entrada da não-linearidade mostrada na Figura 8.2, teremos

$$C_j \frac{dv_j(t)}{dt} + \frac{v_j(t)}{R_j} = \sum_{i=0}^{N-1} w_{ji} x_i(t) + I_j \quad (8.1)$$

O termo capacitivo $C_j dv_j(t)/dt$ no lado esquerdo da Equação (8.1) é a forma mais simples de modelar o comportamento dinâmico de um neurônio biológico visto que, implicitamente, é adicionada memória ao modelo. Dado o campo local induzido $v_j(t)$ pode-se, então, determinar a saída do neurônio j através do uso da relação não-linear

$$x_j(t) = \varphi(v_j(t)) \quad (8.2)$$

O modelo RC descrito pela Equação (8.1) é comumente referido como o modelo aditivo. Esta terminologia é usada para discriminar o modelo aditivo do denominado modelo multiplicativo, onde w_{ji} é dependente de x_i .

Uma característica específica do modelo aditivo descrito pela Equação (8.1) é que o sinal $x_i(t)$ aplicado ao neurônio j pelo neurônio vizinho i é uma função que varia lentamente com o tempo t . Este modelo constitui a base da neurodinâmica clássica.

Consideremos agora uma rede recorrente consistindo da interconexão de N neurônios, cada um deles tendo o mesmo modelo matemático descrito pelas Equações (8.1) e (8.2). Então, ignorando os atrasos devidos ao tempo de propagação entre neurônios (no

contexto de uma rede de neurônios biológicos), pode-se definir a dinâmica da rede pelo seguinte sistema de equações diferenciais de primeira ordem:

$$C_j \frac{dv_j(t)}{dt} = -\frac{v_j(t)}{R_j} + \sum_{i=0}^{N-1} w_{ji} x_i(t) + I_j, \quad j = 0, 1, \dots, N-1 \quad (8.3)$$

o qual resulta da reordenação dos termos de (8.1).

Assume-se que a função de ativação $\varphi(\cdot)$ que relaciona a saída $x_j(t)$ do neurônio j a seu campo local induzido $v_j(t)$ é uma função contínua e, portanto, derivável. Uma função de ativação comumente utilizada é a função logística

$$\varphi(v_j) = \frac{1}{1 + \exp(-v_j)}; \quad j = 0, 1, \dots, N-1 \quad (8.4)$$

Uma condição necessária para que uma Rede de Hopfield seja capaz de aprender é que a rede recorrente descrita pelas Equações (8.2) e (8.3) possua pontos estáveis, denominados **Atratores**.

Reconhecendo que $x_i(t) = \varphi_i(v_i(t))$, pode-se reescrever a (8.3) sob a forma

$$C_j \frac{dv_j(t)}{dt} = -\frac{v_j(t)}{R_j} + \sum_{i=0}^{N-1} w_{ji} \varphi_i(v_i(t)) + I_j, \quad j = 0, 1, \dots, N-1. \quad (8.5)$$

Para prosseguir a discussão, façamos as seguintes considerações:

<p>1. A matriz de pesos sinápticos é simétrica, conforme mostrado por</p> $w_{ji} = w_{ij} \quad \forall i, j \quad (8.6)$
<p>2. Cada neurônio tem uma ativação não-linear própria, daí o uso de $\varphi_i(\cdot)$ na Equação (8.5), no contexto de tentar reproduzir uma rede de neurônios biológicos.</p>
<p>3. O inverso da função de ativação não-linear existe, então pode-se escrever</p> $v = \varphi_i^{-1}(x) \quad (8.7)$

Seja a função sigmóide $\varphi_i(v)$ definida pela função tangente hiperbólica, conforme

$$x = \varphi_i(v) = \tanh\left(\frac{a_i v}{2}\right) = \frac{1 - \exp(-a_i v)}{1 + \exp(-a_i v)}, \quad (8.8)$$

a qual tem uma declividade de $a_i/2$ na origem, conforme mostrado por

$$\frac{a_i}{2} = \left. \frac{d\varphi_i}{dv} \right|_{v=0} \quad (8.9)$$

onde a_i é denominado o **ganho do neurônio i** .

A partir de (8.8), a relação inversa saída-entrada expressa na Equação (8.7) é

$$v = \varphi_i^{-1}(x) = -\frac{1}{a_i} \ln\left(\frac{1-x}{1+x}\right) \quad (8.10)$$

A relação inversa saída-entrada padrão para um neurônio de ganho unitário é

$$\varphi^{-1}(x) = -\ln\left(\frac{1-x}{1+x}\right) \quad (8.11)$$

Podemos, então, reescrever a Equação (8.10) como

$$\varphi_i^{-1}(x) = \frac{1}{a_i} \varphi^{-1}(x) \quad (8.12)$$

A Figura 8.3a mostra o gráfico da não-linearidade sigmoidal padrão $\varphi(v)$, e a Figura 8.3b mostra o gráfico correspondente da não-linearidade inversa $\varphi^{-1}(x)$.

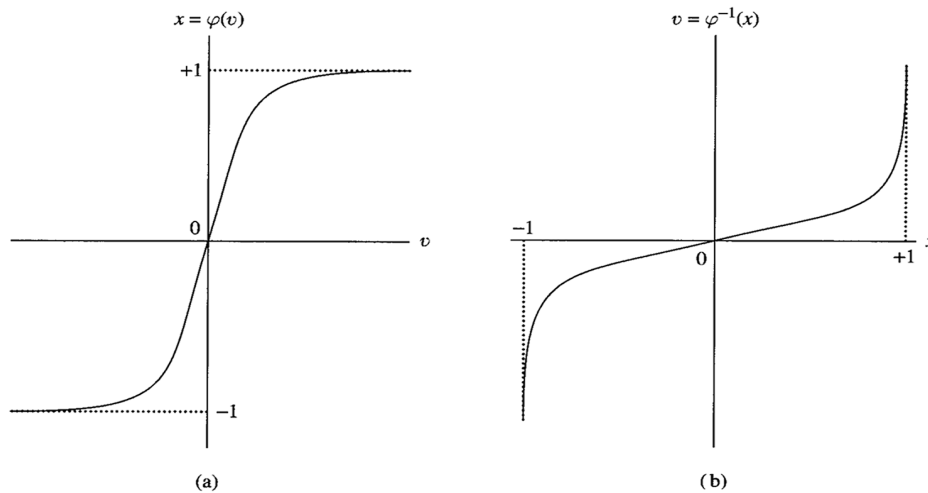


Figura 8.3: Gráficos representativos de (a) não-linearidade sigmoidal padrão e (b) o seu inverso.

No contexto de Estabilidade de Sistemas Dinâmicos, a Função de Energia de Lyapunov [2] da Rede de Hopfield mostrada na Figura 8.1 é dada por

$$E = -\frac{1}{2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} w_{ji} x_i x_j + \sum_{j=0}^{N-1} \frac{1}{R_j} \int_0^{x_j} \varphi_j^{-1}(x) dx - \sum_{j=0}^{N-1} I_j x_j \quad (8.13)$$

A função definida por (8.13) mede a energia “armazenada” no sistema e é uma superfície hiperdimensional que pode apresentar muitos mínimos. A dinâmica da rede é tal que a trajetória do vetor de estado $\underline{x} = [x_0 \ x_1 \ \dots \ x_{N-1}]^T$ (ver Figura 8.1) sempre movimenta-se na direção dos mínimos de energia do sistema (Atratores).

Diferenciando E com respeito ao tempo, teremos

$$\frac{dE}{dt} = -\sum_{j=0}^{N-1} \left(\sum_{i=0}^{N-1} w_{ji} x_i - \frac{v_j}{R_j} + I_j \right) \frac{dx_j}{dt} \quad (8.14)$$

A quantidade entre parênteses, no lado direito da Equação (8.14), é o termo como $C_j dv_j / dt$ da Equação (8.5). Pode-se, então, simplificar a Equação (8.14) para

$$\frac{dE}{dt} = -\sum_{j=0}^{N-1} C_j \left(\frac{dv_j}{dt} \right) \frac{dx_j}{dt} \quad (8.15)$$

Utilizando a relação inversa que define v_j em termos de x_j (Equação 8.7) na Equação (8.15), temos

$$\frac{dE}{dt} = -\sum_{j=0}^{N-1} C_j \left[\frac{d}{dt} \varphi_j^{-1}(x_j) \right] \frac{dx_j}{dt} = -\sum_{j=0}^{N-1} C_j \left(\frac{dx_j}{dt} \right)^2 \left[\frac{d}{dx_j} \varphi_j^{-1}(x_j) \right] \quad (8.16)$$

A partir da Figura 8.3b vemos que a relação saída-entrada inversa $\varphi_j^{-1}(x_j)$ é uma função monotonicamente crescente da saída x_j . Segue, portanto que

$$\frac{d}{dx_j} \varphi_j^{-1}(x_j) \geq 0 \quad \text{para todo } x_j \quad (8.17)$$

Observe também que

$$\left(\frac{dx_j}{dt}\right)^2 \geq 0 \quad \text{para todo } x_j \quad (8.18)$$

Portanto, todos os fatores que constituem o somatório no lado direito da Equação (8.16) são não-negativos. Conseqüentemente, a variação no tempo da função de energia E definida por (8.13), é tal que

$$\frac{dE}{dt} \leq 0$$

A partir da definição da Equação (8.13), notamos que a função E é *bounded* ($t \rightarrow \infty \Rightarrow E \rightarrow E_{\min}$). De acordo com isso, podemos fazer duas afirmações:

1. A função de energia E é uma Função de Energia de Lyapunov do modelo contínuo da Rede de Hopfield.
2. O modelo é estável de acordo com o Teorema 1 de Lyapunov [2].

Em outras palavras, a evolução no tempo do modelo contínuo de Hopfield, descrita pelo sistema de equações diferenciais não-lineares de primeira ordem (Equação (8.5)) representa uma trajetória no espaço de estado, a qual movimenta-se na direção de um dos mínimos da Função de Energia de Lyapunov e que acaba por terminar em um destes pontos. Daí, então, a denominação de Atratores para as coordenadas do vetor de estado $\underline{x} = [x_0 \quad x_1 \quad \dots \quad x_{N-1}]^T$ quando $E(\underline{x}) = E_{\min}$. A partir da Equação (8.16) pode-se observar também que a derivada dE/dt é nula (isto é, o sistema encontra-se em um nível de energia estável) somente se.

$$\frac{d}{dt}x_j(t) = 0 \quad \text{para todo } j$$

Podemos, então, seguir um passo adiante e escrever

$$\frac{dE}{dt} < 0 \quad \text{exceto em um ponto fixo} \quad (8.19)$$

A Equação (8.19) provê base para o seguinte postulado: "A Função de Energia de Lyapunov E de uma Rede Hopfield é uma função monotonicamente decrescente do tempo." Assim, a Rede de Hopfield é global e assintoticamente estável com os mínimos de E ocorrendo para coordenadas do vetor de estado $\underline{x} = [x_0 \ x_1 \ \dots \ x_{N-1}]^T$ que correspondam às coordenadas dos Atratores da rede.

8.1 Relação entre os Estados Estáveis das Versões Contínua e Discreta do Modelo de Hopfield

A rede de Hopfield pode ser operada em um modo contínuo ou discreto, dependendo do modelo adotado para descrever os neurônios.

O modo contínuo de operação é baseado em um modelo aditivo, conforme previamente descrito. O modo discreto de operação é baseado no modelo de McCulloch-Pitts.

Podemos prontamente estabelecer a relação entre os estados estáveis do modelo contínuo de Hopfield e os estados estáveis do modelo discreto de Hopfield correspondente, através da redefinição da relação entrada-saída para um neurônio que satisfaça a duas simplificações:

1. A saída de um neurônio tem os valores assintóticos

$$x_j = \begin{cases} +1 & \text{para } v_j = \infty \\ -1 & \text{para } v_j = -\infty \end{cases} \quad (8.20)$$

2. O ponto médio da função de ativação de um neurônio fica na origem, conforme mostrado por

$$\varphi_j(0) = 0 \quad (8.21)$$

De forma correspondente, pode-se fazer o *bias* I_j igual a zero para todo j .

Ao formular a função de energia E para um modelo contínuo de Hopfield, é permitido que os neurônios tenham laços de auto-realimentação. Um modelo discreto de Hopfield, por outro lado, não necessita ter laços de auto-realimentação. Pode-se, portanto, simplificar a discussão fazendo $w_{jj} = 0$ para todos j em ambos os modelos.

A partir destas observações, podemos redefinir a função de energia de um modelo contínuo de Hopfield dada pela Equação (8.13) conforme segue:

$$E = -\frac{1}{2} \sum_{i=0}^{N-1} \sum_{\substack{j=0 \\ i \neq j}}^{N-1} w_{ji} x_i x_j + \sum_{j=0}^{N-1} \frac{1}{R_j} \int_0^{x_j} \varphi^{-1}(x) dx \quad (8.22)$$

A função inversa $\varphi^{-1}(x)$ é definida pela Equação (8.12). Podemos então reescrever a função de energia da Equação (8.22), conforme segue:

$$E = -\frac{1}{2} \sum_{i=0}^{N-1} \sum_{\substack{j=0 \\ i \neq j}}^{N-1} w_{ji} x_i x_j + \sum_{j=0}^{N-1} \frac{1}{a_j R_j} \int_0^{x_j} \varphi^{-1}(x) dx \quad (8.23)$$

A integral

$$\int_0^{x_j} \varphi^{-1}(x) dx$$

tem a forma padrão mostrada na Figura 8.4. Seu valor é zero para $x_j = 0$ e positivo em todos os outros casos. Ainda, a integral assume um valor muito grande à medida que x_j se aproxima de ± 1 .

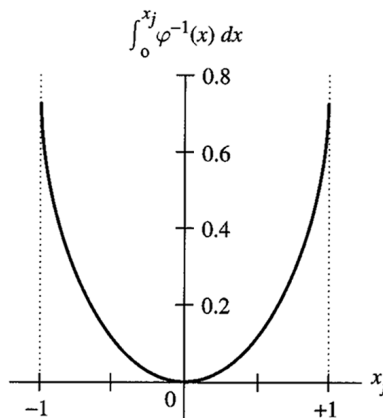


Figura 8.4: Gráfico representativo da integral $\int_0^{x_j} \varphi^{-1}(x) dx$.

Se, entretanto, o ganho a_j do neurônio j se torna infinitamente grande (isto é, a não-linearidade sigmoidal se aproxima da função de transferência de um *hard limiter*), o segundo termo da Equação (8.23) se torna desprezivelmente pequeno. No caso limite, quando $a_j = \infty$ para todo j , os valores máximo e mínimo das componentes do vetor de estado $\underline{x} = [x_0 \ x_1 \ \dots \ x_{N-1}]^T$ no modelo contínuo de Hopfield se tornam idênticos àqueles no modelo discreto de Hopfield. Para o caso do modelo discreto, a Função de Energia de Lyapunov é definida simplesmente por

$$E = -\frac{1}{2} \sum_{i=0}^{N-1} \sum_{\substack{j=0 \\ i \neq j}}^{N-1} w_{ji} x_i x_j \quad (8.24)$$

onde o estado do j -ésimo neurônio assume os valores $x_j = \pm 1$. Conclui-se, portanto, que os pontos estáveis (Atratores) do modelo determinístico contínuo de alto ganho de Hopfield correspondem aos pontos estáveis do modelo discreto estocástico de Hopfield.

Quando, entretanto, cada neurônio j tem um ganho a_j grande, porém finito, o segundo termo do lado direito da Equação (8.23) assume uma contribuição não insignificante para a função de energia do modelo contínuo. Em particular, esta contribuição é grande e positiva próximo a todas as faces, bordas e cantos do hipercubo unitário que define o espaço de estados do modelo. Por outro lado, a contribuição é desprezivelmente pequena em pontos que estão distantes das faces. De acordo com isso, a função de energia de tal modelo tem seu máximo nos cantos, mas os mínimos estão levemente posicionados em direção ao interior do hipercubo. A Figura 8.5 mostra o mapa dos Contornos de Energia (*energy landscape*) para um modelo contínuo de Hopfield usando dois neurônios.

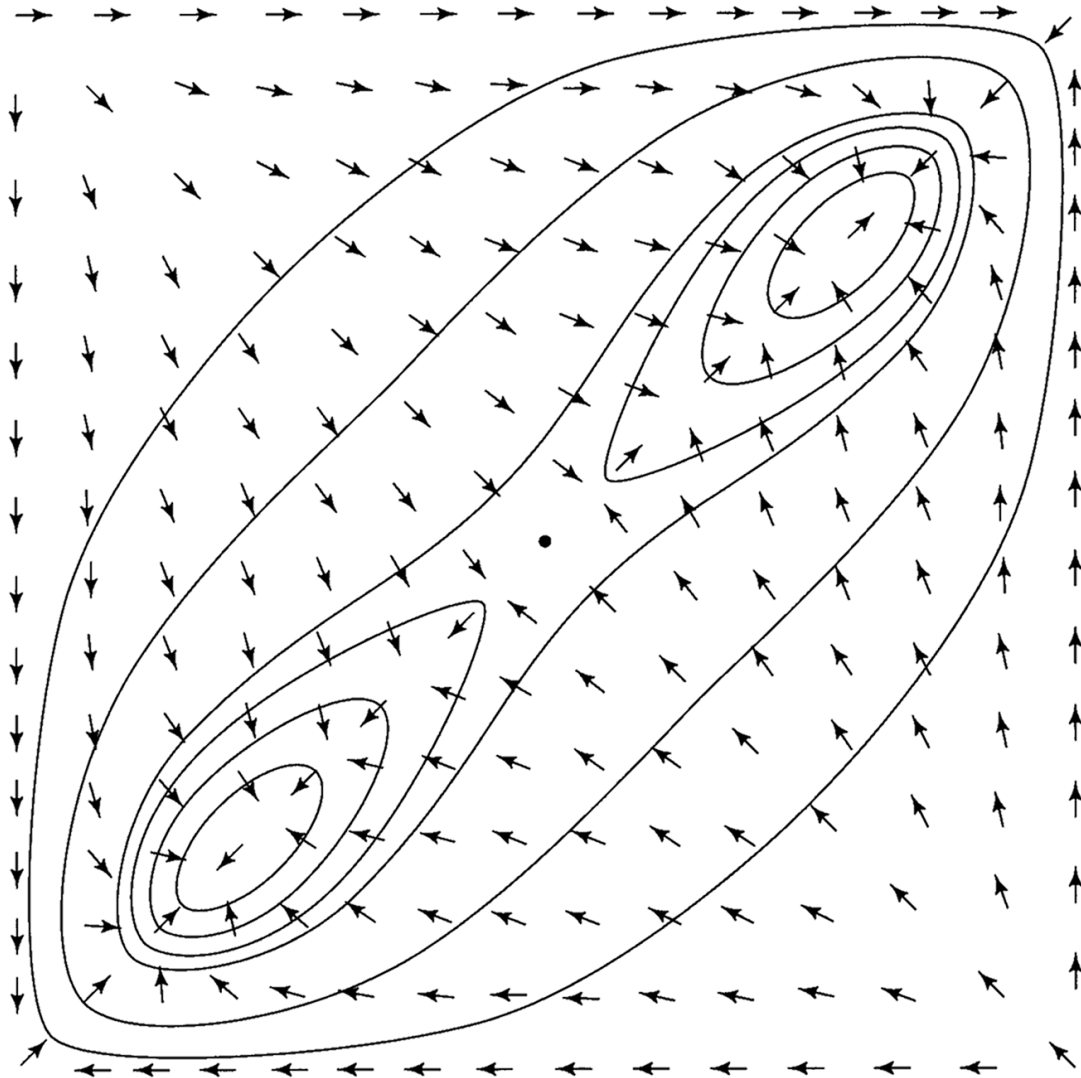


Figura 8.5: Mapa dos Contornos de Energia (linhas sólidas) para uma Rede de Hopfield composta de dois neurônios. A ordenada x_1 e a abscissa x_0 são as componentes do vetor de estado $\underline{x} = [x_0 \ x_1]^T$, correspondendo respectivamente a cada um dos nós de saída dos dois neurônios que compõe a rede. Os dois Atratores desta rede estão localizados próximo ao canto esquerdo inferior e canto direito superior da figura. Os pontos instáveis localizam-se nos outros dois cantos. As setas definem a trajetória de movimentação de $\underline{x} = [x_0 \ x_1]^T$ (*state flow*) a partir de qualquer coordenada inicial no mapa. O *state flow* na direção dos Atratores é consequência do processo de minimização da Função de Energia de Lyapunov, dada por (8.13), efetuado pela Rede de Hopfield a medida que o tempo t transcorre a partir de um estado inicial $\underline{x}(t) = [x_0(t) \ x_1(t)]^T \Big|_{t=0} = \underline{x}(0) = [x_0(0) \ x_1(0)]^T$. Observe que cada Atrator constitui um sumidouro (*sink*) para o campo vetorial formado pelas setas do *state flow*.

8.2 O Modelo Discreto de Hopfield Visto como uma *Content-Addressable Memory* (CAM).

As Redes de Hopfield têm atraído muita atenção na literatura como CAM. Neste tipo de aplicação, sabemos *a priori* os Atratores da rede, no sentido de que eles correspondem a padrões a serem armazenados. Entretanto, os pesos sinápticos da rede que produzem os Atratores são desconhecidos, e o problema é como determiná-los.

A função primordial de uma CAM é recuperar um padrão armazenado na memória, em resposta à apresentação de uma versão incompleta ou ruidosa daquele padrão. Para ilustrar o significado desta afirmativa, de uma forma sucinta, nada melhor podemos fazer do que citar o próprio Hopfield (1982):

“Suponha que um item armazenado na memória seja “H. A. Kramers & G. H. Wannier *Physi Rev.*60, 252 (1941).” Uma CAM deve ser capaz de recuperar este item inteiro da memória com base em uma informação parcial. A entrada “& Wannier (1941)” deve ser suficiente. Uma memória ideal pode lidar com erros e recuperar esta referência mesmo a partir da entrada “Wannier, (1941).”

Uma propriedade importante de uma CAM é, portanto, a habilidade de recuperar um padrão armazenado, dado um subconjunto razoável do conteúdo de informação daquele padrão. Mais ainda, uma CAM permite correção de erro no sentido de que ela pode ignorar informações inconsistentes nas sugestões apresentadas a ela.

A essência de uma CAM é mapear uma memória fundamental $\underline{\xi}_\mu$ em um Atrator \underline{x}_μ de um sistema dinâmico, conforme ilustrado na Figura 8.6. Matematicamente, podemos expressar este mapeamento na forma $\underline{\xi}_\mu \Leftrightarrow \underline{x}_\mu$.

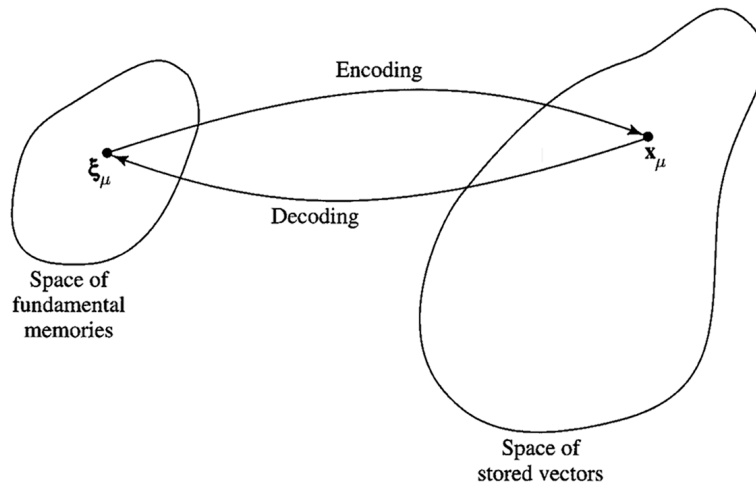


Figura 8.6: Ilustração da codificação-decodificação desempenhada por uma rede recorrente.

A seta da esquerda para a direita descreve uma operação de codificação, enquanto que a seta da direita para a esquerda descreve uma operação de decodificação. Os pontos fixos atratores do espaço de estado da rede são as memórias fundamentais ou estados protótipos da rede.

Suponha agora que seja apresentado à rede um padrão contendo informação parcial mas suficiente sobre uma das memórias fundamentais. Podemos então representar aquele particular padrão como ponto de partida no espaço de estados. Em princípio, desde que o ponto de partida (isto é, o estado inicial $\underline{x}(t) = [x_0(t) \ x_1(t)]^T \Big|_{t=0} = \underline{x}(0) = [x_0(0) \ x_1(0)]^T$) seja próximo ao Atrator que representa a memória a ser recuperada (isto é, desde que $\underline{x}(0)$ esteja dentro da bacia de atração de um dos Atratores), o sistema deve evoluir com o tempo e finalmente convergir para o próprio Atrator que representa a memória. Neste instante de tempo a memória inteira é recuperado pela rede. Conseqüentemente, a Rede de Hopfield tem uma propriedade emergente, a qual a auxilia a recuperar informação e a lidar com erros.

Com o modelo de Hopfield utilizando o neurônio formal de McCulloch e Pitts (1943) como unidade básica de processamento, cada neurônio tem dois estados determinados pelo nível do campo local induzido que atua sobre ele. O estado “ligado” do

neurônio i é denotado pela saída $x_i = +1$, e o estado “desligado” é representado por $x_i = -1$. Para uma rede constituída de N neurônios, o estado da rede é então definido pelo vetor $\underline{x} = [x_0 \ x_1 \ \dots \ x_{N-1}]^T$.

Com $x_i = \pm 1$, o estado do neurônio i representa um *bit* de informação, e o vetor de estado $\underline{x} \in \mathfrak{R}^N$, representa uma palavra binária de N *bits* de informação.

O campo induzido local v_j do neurônio j é definido por

$$v_j = \sum_{i=0}^{N-1} w_{ji} x_i + b_j \quad (8.25)$$

onde b_j é um *bias* fixo aplicado externamente ao neurônio j . Portanto, o neurônio j modifica seu estado x_j de acordo com (neurônio de McCulloch e Pitts):

$$x_j = \begin{cases} +1 & \text{se } v_j > 0 \\ -1 & \text{se } v_j < 0 \end{cases}$$

Esta relação pode ser reescrita na forma compacta

$$x_j = \text{sgn}[v_j]$$

onde $\text{sgn}(\cdot)$ retorna o sinal do argumento. Se v_j é zero, o neurônio j permanece em seu estado anterior, desconsiderando se está ligado ou desligado. O significado de assumir esta convenção é que o diagrama de fluxo resultante é simétrico, como será ilustrado posteriormente.

Há duas fases para a operação da rede discreta de Hopfield como uma CAM, chamadas:

- Armazenagem (*Storage phase*) e
- Recuperação (*Retrieval phase*).

1. Storage

Suponha que desejamos armazenar um conjunto de vetores N -dimensionais (palavras binárias), denotado por $\{\underline{\xi}_\mu \mid \mu = 0, 1, \dots, M-1\}$. Chamamos estes M vetores de memórias fundamentais, representando os padrões a serem memorizados pela rede.

Seja $\xi_{\mu,i}$ o i -ésimo elemento da memória fundamental $\underline{\xi}_\mu$.

De acordo com a regra do produto externo de armazenamento, ou seja, a generalização do postulado de aprendizado de Hebb, o peso sináptico do neurônio i para o neurônio j é definido por

$$w_{ji} = \frac{1}{N} \sum_{\mu=0}^{M-1} \xi_{\mu,j} \xi_{\mu,i} \quad (8.26)$$

A motivação para usar $1/N$ como constante de proporcionalidade é simplificar a descrição matemática da recuperação da informação. Note também que a regra de aprendizado da Equação (8.26) é um procedimento não-iterativo, isto é, os pesos sinápticos w_{ji} ficam totalmente determinados após a soma de M termos.

Observe que o que significa os neurônios da Rede de Hopfield não têm auto-realimentação (ver Figura 8.1), de modo que

$$w_{ii} = 0 \quad \text{para todo } i \quad (8.27)$$

Seja \mathbf{W} a matriz de pesos sinápticos de dimensão $[N \times N]$ da rede, com w_{ji} sendo seu ji -ésimo elemento. Podemos então combinar as Equações (8.26) e (8.27) em uma única equação escrita na forma matricial, conforme segue:

$$\mathbf{W} = \frac{1}{N} \sum_{\mu=0}^{M-1} \underline{\xi}_\mu \underline{\xi}_\mu^T - \mathbf{M}\mathbf{I} \quad (8.28)$$

onde $\underline{\xi}_\mu \underline{\xi}_\mu^T$ representa o produto externo do vetor $\underline{\xi}_\mu$ com ele próprio, e \mathbf{I} denota a matriz identidade. A partir destas definições dos pesos sinápticos/matriz de pesos, podemos, mais uma vez, verificar que:

- A saída de cada neurônio na rede é realimentada para as entradas de todos os outros neurônios.
- Não há elo de auto-realimentação na rede (isto é, $w_{ii} = 0$).
- A matriz de pesos da rede é simétrica (ver Equação (8.6)), isto é,

$$\mathbf{W}^T = \mathbf{W} \quad (8.29)$$

2. Retrieval

Durante a Recuperação, um vetor N -dimensional $\underline{\zeta}_{sonda}$ é imposto à Rede de Hopfield como seu estado. O vetor sonda tem os seus elementos iguais a ± 1 . Tipicamente, o vetor representa uma versão incompleta ou ruidosa de uma memória fundamental da rede.

A recuperação de informação acontece de acordo com uma regra dinâmica iterativa, na qual, a cada iteração, um neurônio j da rede é aleatoriamente escolhido e é examinado o campo local induzido v_j (incluindo qualquer *bias* não nulo b_j).

Se, a qualquer instante de tempo $v_j > 0$ o neurônio j irá trocar seu estado para $+1$, ou permanecerá naquele estado se já estiver lá.

De forma similar, se $v_j < 0$, o neurônio j irá trocar seu estado para -1 , ou permanecerá neste estado se já estiver lá.

Se $v_j = 0$, o neurônio j é deixado no seu estado prévio, não importando se estava ligado ($x_j = +1$) ou desligado ($x_j = -1$).

A atualização do estado de uma iteração para a próxima é, portanto, determinística, mas a seleção de um neurônio para que seja executada a atualização é aleatória.

O procedimento de atualização assíncrona (serial) aqui descrito é continuado até que não haja mais mudanças a reportar. Isto é, começando com o vetor sonda \underline{x} , a rede finalmente produz um vetor de estado \underline{y} invariante no tempo, cujos elementos individuais satisfazem a condição para a estabilidade:

$$y_i = \text{sgn} \left(\sum_{j=0}^{N-1} w_{ji} y_j + b_j \right), \quad j = 1, 2, \dots, N \quad (8.30)$$

ou, sob a forma matricial,

$$\underline{y} = \text{sgn} (\mathbf{W} \underline{y} + \underline{b}), \quad (8.31)$$

onde \mathbf{W} é a matriz de pesos sinápticos da rede, e \underline{b} é o vetor *bias* externamente aplicado.

A condição de estabilidade aqui descrita é também referida como a condição de alinhamento. O vetor de estado \underline{y} que a satisfaz é chamado um estado estável ou Atrator do espaço de estados do sistema.

Podemos, portanto, afirmar que a Rede de Hopfield irá sempre convergir para um estado estável quando a operação de recuperação é conduzida de forma assíncrona.

8.3 Sumário do Modelo de Hopfield

A Tabela 8.1 apresenta o sumário dos passos envolvidos nas fases operacionais de Armazenagem e de Recuperação de uma Rede de Hopfield.

O passo 1 é a fase de Armazenagem e os passos 2 a 4 constituem a fase de Recuperação.

1. Aprendizado

Seja $\{\underline{\xi}_0, \underline{\xi}_1, \dots, \underline{\xi}_{M-1}\}$ um conjunto conhecido de memórias N -dimensionais fundamentais (Atratores).

→ Usa-se a regra do produto externo (isto é, o postulado de aprendizagem de Hebb) para computar os pesos sinápticos da rede:

$$w_{ji} = \begin{cases} \frac{1}{N} \sum_{\mu=0}^{M-1} \xi_{\mu,j} \xi_{\mu,i} & j \neq i \\ 0, & j = i \end{cases}$$

Onde w_{ji} é o peso sináptico do neurônio i para o neurônio j .

Os elementos do vetor $\underline{\xi}_\mu$ são iguais a ± 1 .

Uma vez determinados os pesos sinápticos w_{ji} , estes são mantidos fixos.

2. Inicialização

Seja $\underline{\xi}_{sonda}$ um vetor de entrada N -dimensional desconhecido (sonda) apresentado à rede.

→ O algoritmo é inicializado fazendo-se

$$x_j(0) = \xi_{j,sonda} \quad \text{com} \quad j = 0, 1, \dots, N-1.$$

Onde x_j é o estado do neurônio j no tempo $n = 0$, e

$\xi_{j,sonda}$ é o j -ésimo elemento do vetor sonda $\underline{\xi}_{sonda}$.

3. Iteração até a convergência

- Os elementos do vetor de estados $\underline{x}(n)$ são atualizados de forma assíncrona (isto é, aleatoriamente e um de cada vez), de acordo com a regra

$$x_j(n+1) = \text{sgn} \left[\sum_{i=0}^{N-1} w_{ji} x_i(n) \right], \quad j = 0, 1, \dots, N-1.$$

- A iteração é repetida até que o vetor de estado \underline{x} permaneça inalterado.

4. Saída

Seja \underline{x}_{sink} o vetor que representa as coordenadas do Atrator computado ao final do passo três.

- O resultante vetor de saída \underline{y} da rede é dado por

$$\underline{y} = \underline{x}_{sink}$$

8.4 Referências Bibliográficas do Capítulo 8

- [1] S. Haykin, *Neural Networks*, 2nd ed., Prentice Hall, Upper Saddle River, New Jersey, 1999.
- [2] C. T. Chen, *Linear System Theory and Design*, Harcourt Brace College Publishers, 1984.
- [3] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*, MIT Press, 1995.
- [4] J. Hertz, A. Krogh e R. G. Palmer, *Introduction to the Theory of Neural Computation*, Addison-Wesley, 1991.

Apêndice: Rede de Hopfield - Codificação Não - Linear

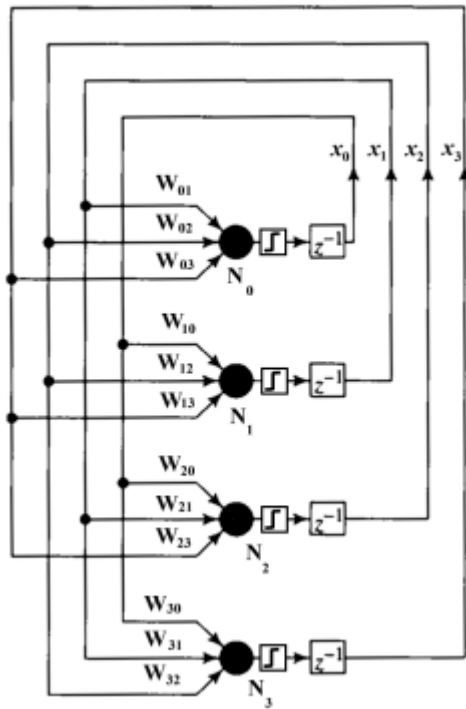


Figura 1: Rede de Hopfield com 4 neurônios e respectivos *hard limiters*.

Funções auxiliares deste *script* para MathCad 7.0:

$$\text{Bin}(\text{HopfieldVector}) := \frac{(\text{HopfieldVector} + 1)}{2}$$

$$\text{Hop}(\text{BinWord}) := 2 \cdot \text{BinWord} - 1$$

$$\text{Not}(\text{BinWord}) := 1 - \text{BinWord}$$

$$\text{SpectrDist}(\text{Code}) := \begin{array}{l} \text{nn} \leftarrow \text{cols}(\text{Code}) \\ \text{CodSize} \leftarrow \text{rows}(\text{Code}) \\ \text{cc} \leftarrow 0 \\ \text{for } \text{ia} \in 0.. \text{CodSize} - 2 \\ \quad \text{for } \text{ib} \in \text{ia} + 1.. \text{CodSize} - 1 \\ \quad \quad \text{acc} \leftarrow 0 \\ \quad \quad \text{for } \text{j} \in 0.. \text{nn} - 1 \\ \quad \quad \quad \text{acc} \leftarrow \text{acc} + 1 \text{ if } \text{Code}_{\text{ia},\text{j}} \neq \text{Code}_{\text{ib},\text{j}} \\ \quad \quad \text{Dist}_{\text{cc}} \leftarrow \text{acc} \\ \quad \quad \text{cc} \leftarrow \text{cc} + 1 \\ \text{Dist} \end{array}$$

⇒ O objetivo deste estudo é investigar a capacidade da Rede de Hopfield codificar e decodificar (no contexto de Códigos Para Correção de Erros) um espaço de vetores binários (exceto o vetor $\underline{0}$, caracterizando a Codificação Não-Linear).

Seja, por exemplo, o espaço de vetores binários de dimensão 4 (exceto o vetor $\underline{0}$):

$$\text{BinSpace} := \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Normalizando as componentes dos vetores para +1, -1:

$\xi_0 := \text{Hop}(\text{BinSpace})$

$$\xi_0 = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}$$

Obtendo a matriz de pesos sinápticos W da Rede de Hopfield para o conjunto de vetores ξ_0 :

Nota: W_{ij} é o peso sináptico que conecta a saída do neurônio i à respectiva entrada no neurônio j

$$\text{WeightMatrix}(\xi) := \begin{array}{l} N \leftarrow \text{rows}(\xi) \\ M \leftarrow \text{cols}(\xi) \\ W \leftarrow \frac{1}{N} \cdot \left(\sum_{i=0}^{M-1} \xi^{<i>} \cdot \xi^{<i>T} - M \cdot \text{identity}(N) \right) \\ \text{return } W \end{array}$$

$$W := \text{WeightMatrix}(\xi_0) \quad W = \begin{bmatrix} 0 & -0.25 & -0.25 & -0.25 \\ -0.25 & 0 & -0.25 & -0.25 \\ -0.25 & -0.25 & 0 & -0.25 \\ -0.25 & -0.25 & -0.25 & 0 \end{bmatrix}$$

Obtendo os estados estáveis (Atratores) da Rede de Hopfield para a matriz de pesos sinápticos W obtida do conjunto de vetores ξ_0 :

Nota: Um vetor de estado \underline{x} é estável se $\text{sgn}(\underline{x}W) = \underline{x}$, (Condição de Alinhamento) onde $\text{sgn}(u)$ representa o *hard limiter*.

```

StableStates( $\xi, W$ ) :=
  M ← cols( $\xi$ )
  N ← rows( $\xi$ )
  i ← 0
  for j ∈ 0.. M - 1
    Y ← W ·  $\xi^{<j>}$ 
    for k ∈ 0.. N - 1
      Sk ← 0
      Sk ← -1 if Yk < 0
      Sk ← 1 if Yk > 0
    if [  $\sum (S - \xi^{<j>}) = 0$  ]
      S<i> ← S
      i ← i + 1
  return Ss

```

$$Ss := \text{StableStates}(\xi_0, W) \quad \text{Bin}(Ss) = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Verificando a consistência dos Atratores:

Nota: Se um conjunto Ss de vetores de estado \underline{x} é um conjunto de Atratores então os seus estados estáveis são formados pelos próprios vetores de Ss :

$$W := \text{WeightMatrix}(Ss) \quad W = \begin{bmatrix} 0 & -0.5 & -0.5 & -0.5 \\ -0.5 & 0 & -0.5 & -0.5 \\ -0.5 & -0.5 & 0 & -0.5 \\ -0.5 & -0.5 & -0.5 & 0 \end{bmatrix} \quad \text{Bin}(\text{StableStates}(Ss, W)) = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Calculando as distâncias de Hamming entre os Atratores:

Nota: Quanto maior a Distância de Hamming mínima maior a capacidade de correção de erro do decodificador.

$$\text{SpectrDist}(\text{Bin}(\text{Ss})^T) = [2 \ 2 \ 2 \ 2 \ 4 \ 2 \ 2 \ 4 \ 2 \ 4 \ 2 \ 2 \ 2 \ 2 \ 2]$$

Resultado que não é muito satisfatório. Experimentemos determinar a Distância de Hamming entre os Atratores, só que TRANSPOSTOS, isto é:

$$\text{Bin}(\text{Ss}^T) = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

$$\text{SpectrDist}(\text{Ss})^T = [4 \ 4 \ 4 \ 4 \ 4 \ 4]$$

O que é um resultado EXTREMAMENTE satisfatório, já que conseguimos maximizar a Distância de Hamming entre todos os vetores . Portanto estes vetores são palavras código binárias com bom potencial para codificação. Ainda é necessário verificar se os Atratores transpostos formam um conjunto de Atratores:

$$\xi := \text{Ss}^T \quad \text{Bin}(\xi) = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \quad \text{W} := \text{WeightMatrix}(\xi) \quad \text{W} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -0.667 \\ 0 & 0 & 0 & 0 & -0.667 & 0 \\ 0 & 0 & 0 & -0.667 & 0 & 0 \\ 0 & 0 & -0.667 & 0 & 0 & 0 \\ 0 & -0.667 & 0 & 0 & 0 & 0 \\ -0.667 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\text{Ss_New} := \text{StableStates}(\xi, \text{W}) \quad \text{Bin}(\text{Ss_New}) = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix} \quad \text{Bin}(\xi) = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Portanto, os Atratores transpostos também formam um conjunto de Atratores.

No entanto, a Rede de Hopfield pode convergir para outros Atratores diferentes daqueles definidos pela matriz W obtida do conjunto de vetores ξ . Tais Atratores são denominados **Atratores Espúrios (estados espúrios)**:

No presente caso, o conjunto de Atratores Espúrios ξ_{espur} são os vetores obtidos da negação lógica (NOT) das componentes do conjunto de Atratores ξ :

$$\xi_{\text{espur}} := \text{Hop}(\text{Not}(\text{Bin}(\xi))) \quad \text{Bin}(\xi_{\text{espur}}) = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad \text{Bin}(\xi) = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Observe que o conjunto de Atratores Espúrios ξ_{espur} também obedece a **Condição de Alinhamento** $\text{sgn}(\underline{x}W) = \underline{x}$, onde \underline{x} é um dos vetores do conjunto ξ_{espur} :

$$\text{Bin}(\text{StableStates}(\xi_{\text{espur}}, W)) = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad W = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -0.667 \\ 0 & 0 & 0 & 0 & -0.667 & 0 \\ 0 & 0 & 0 & -0.667 & 0 & 0 \\ 0 & 0 & -0.667 & 0 & 0 & 0 \\ 0 & -0.667 & 0 & 0 & 0 & 0 \\ -0.667 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Os Atratores Espúrios ξ_{espur} decorrem da situação em que W possui autovalores muito próximos ou iguais (nulos ou não), como é o caso em questão, para o qual temos 2 grupos de 3 autovalores iguais:

$$\text{eigenvals}(W)^T = [-0.667 \quad 0.667 \quad -0.667 \quad 0.667 \quad -0.667 \quad 0.667]$$

Nota: Atratores Espúrios ocorrem também para o conjunto original ξ_0 :

$$\text{eigenvals}(\text{WeightMatrix}(\xi_0))^T = [0.25 \quad 0.25 \quad 0.25 \quad -0.75]$$

Verificando a capacidade de correção de erro da Rede de Hopfield no contexto de Codificação/Decodificação:

Nota: O pseudo-código Retrieval(.), abaixo especificado, implementa a fase de Recuperação de uma Rede de Hopfield com matriz de pesos sinápticos W a partir do vetor sonda \underline{x} . O argumento CheckWindow especifica o número de iterações durante as quais \underline{x} deve permanecer constante para que seja considerado que a rede convergiu para um Atrator.

```

Retrieval(x, W, CheckWindow) :=
    MaxIter ← 1000
    N ← rows(x)
    xnext<0> ← x
    for n ∈ 1.. MaxIter - 1
        j ← floor(rnd(N))
        vj ← ∑i=0N-1 Wi,j · xi
        xj ← -1 if vj < 0
        xj ← 1 if vj > 0
        xnext<n> ← x
        Test ← ∑k=0CheckWindow [ ∑ (xnext<n-k> - xnext<n-k-1>) ] if n > CheckWindow
        return xnext<n> if Test = 0
    return xnext

```


Codificando e decodificando:

$$\text{Bin}(Ss) = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix} \quad \text{Not}(\text{Bin}(Ss)) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Observe do conjunto de vetores Ss acima que a Rede de Hopfield em questão gera um código binário **NÃO LINEAR** (não existe a palavra-código $\underline{0}$) e **Sistemático** com mensagens de $k=2$ bits e palavras-código resultantes de $n=6$ bits.

Vamos supor que a Rede de Hopfield receba a seguinte palavra-código:

$$\text{Received} := (1 \ 1 \ 0 \ 1 \ 1 \ 0)^T$$

A rede pode convergir para o Atrator mais próximo em termos da Distância de Hamming:

$$\text{Decoded} := \text{Bin}(\text{Retrieval}(\text{Hop}(\text{Received}), W, 10\text{-rows}(Ss))) \quad \text{Decoded}^T = [1 \ 1 \ 0 \ 1 \ 0 \ 0]$$

Ou pode convergir para o Atrator Espúrio mais próximo em termos da Distância de Hamming:

$$\text{Decoded} := \text{Bin}(\text{Retrieval}(\text{Hop}(\text{Received}), W, 10\text{-rows}(Ss))) \quad \text{Decoded}^T = [1 \ 0 \ 0 \ 1 \ 1 \ 0]$$

Uma possível solução para a ambigüidade gerada pelos Atratores Espúrios é, ocorrendo convergência para um estado estável que não pertence ao conjunto de Atratores ξ , executa-se novamente o processo até que ocorra convergência para um estado estável pertencente a ξ .

Uma outra possível solução para a ambigüidade gerada pelos Atratores Espúrios é conduzir as iterações na fase de *Retrieval* da Rede de Hopfield de modo que, se a distância de Hamming entre o vetor de estado \underline{x} e os vetores do conjunto de Atratores Espúrios ξ_{espur} cair abaixo de um valor d_{min} considerado seguro, a iteração é descartada e o vetor \underline{x} não é atualizado.