LABIC

## Redes Neurais Auto-organizáveis
## Modelo de Hopfield

Profa. Roseli Ap. Francelin Romero



LABIC

## Modelo de Hopfield (1982)

- O problema de Memória Associativa

"Armazene um conj. de p padrões $x_i^\lambda$ de tal forma que quando uma nova entrada $x_i$ é apresentada, a rede responde produzindo um dos padrões que melhor se parecem com $x_i$".

Vamos supor que os padrões são nomeados por $\lambda$ = 1,2,...,p enquanto que os nós na rede são nomeados por i=1,2,...,N.

Ambos os padrões $x_i^\lambda$ e $x_i$ são constituídos de valores -1 e 1.



LABIC

## Modelo de Hopfield

- É claro que este probl. pode ser implementado num computador serial, simplesmente armazenando uma lista de padrões $x_i^\lambda$, escrevendo um programa que calcule a distância de Hamming (no. bits diferentes) entre o padrão teste $x_i$ e cada dos padrões armazenados, encontrando qual deles possui a menor dist. de Hamming $x_i$ e imprimindo o correspondente padrão armazenado.



LABIC

## Modelo de Hopfield

- Objetivo: Mostrar como uma rede constituída de neurônios de McCullock-Pitts pode fazer isto.
- Partir de uma configuração inicial: $\gamma_i = x_i$

  deseja-se mostrar qual conjunto (se existir algum) de $w_{ij}$ fará a rede ir para o estado $\gamma_i = x_i^\lambda$ que é o padrão de menor distância de Hamming de $x_i$
- O que se deseja é que a MEMÓRIA seja endereçável por conteúdo e insensível a erros pequenos no padrão de entrada.



LABIC

## Modelo de Hopfield

- Importância:

  memória endereçável por conteúdo pode ser de grande valia.

  Suponha, por exemplo, que informação codificada sobre muitos cientistas famosos são armazenadas numa rede. Então,

  padrão "evolução" ===> Darwin e

  padrão "E=mc3" ====> Einstein



LABIC

## Modelo de Hopfield

- OBS: sempre um padrão será recuperado ( a menos que se inclua um padrão "não sei").

  A rede numa recuperará uma combinação linear de Darwin com Einsten, em resposta à entrada:"evolução", mas o

  melhor matching de acordo com o que foi armazenado.

LABIC

# Modelo de Hopfield

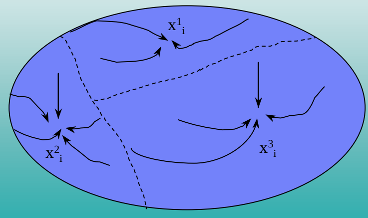
- Outros exs. Memória Associativa
  - reconhecimento e reconstrução de imagens
  - recuperação de informação bibliográfica de referências parciais.



LABIC

# Modelo de Hopfield

- Espaço de configuração de um modelo com 3 atratores






LABIC

# Modelo de Hopfield

- O modelo de Hopfield foi proposto em 1982
- Função de Transferência (Mem. Assoc.):

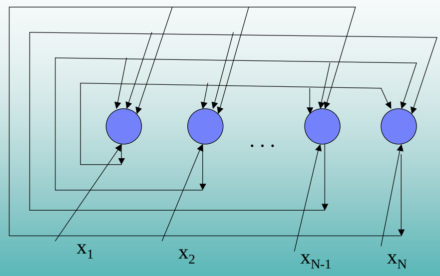
$$f(x) = \begin{cases} 1 & x \geq 0 \\ -1 & \text{caso contrário} \end{cases}$$

que é chamada função sinal.

- A saída de cada neurônio:

$s_j = \Sigma_i \; w_{ji} \, s_i$ onde os thresholds = 0






LABIC

# Modelo de Hopfield

- No caso deste modelo os nós são atualizados assincronamente. O modo sincronizado requer um clock central e é muito sensível a erros de tempo.
- Assíncrono:
  - seleciona-se aleatoriamente um nó (ou unidade) i para ser atualizado. Simulação
  - cada unidade se atualiza independentemente de acordo com uma constante de probabilidade por unidade de tempo. Unidades de Hardware Autônomas



LABIC

# Modelo de Hopfield

- Será assumido que uma configuração estável da rede é obtida quando todos os $S_i$ não sofrem mais alterações.
- TESTE: se um conj. de pesos é aceitável:

  se os padrões memorizáveis são por si só estáveis, então verificar se desvios pequenos nesses padrões são corrigidos quando a rede evolui.

LABIC

## Modelo de Hopfield

- Para motivar a escolha, consideremos o caso mais simples onde existe apenas um padrão

$x=(x_1, ..., x_N)$, para ser memorizado, $x_i$ = -1 ou $x_i$ = 1.

- A condição para que este padrão ser estável:

$$\text{sinal } (\Sigma_i\, w_{ji}\, x_i) = x_j \quad \text{para todo } j$$

o que significa que não houve mudança.

Então:

$$w_{ji} \;\alpha\; x_j\, x_i \qquad \text{desde que } x_i^2 = 1$$



LABIC

## Modelo de Hopfield

- A cte de proporcionalidade, por conveniência é tomada igual a 1/N, onde N é o no. de nós da rede.
- $w_{ji} = (1/N)\; x_j\; x_i$
- Uma configuração próxima (segundo a distância de Hamming) a $x_j$ rapidamente convergirá para $x_j$
- Isto significa que a rede corrigirá erros como desejado e pode-se dizer que o padrão $x_j$ é um atrator. Existem 2 atratores: o - $x_j$ também é um atrator.



LABIC

## Modelo de Hopfield

- Determinação dos pesos para recuperação de vários padrões
  - como conseguir que o sistema recupere o mais similar à entrada entre muitos padrões? A resposta mais simples é tomar $w_{ji}$ similar ao caso anterior:

$w_{ji} = (1/N)\; \Sigma_{\lambda}\, x_j^{\lambda}\, x_i^{\lambda}$ onde $\lambda$ = 1, 2, ...., $p^{\lambda}$

p - no. de padrões

REGRA DE HEBB OU REGRA DE HEBB GENERALIZADA



LABIC

- $w_{ii}$ = 0 para todo i
- W é uma matriz NxN.
- Da eq. (1) e (2) podemos escrever:

$$W = (1/N)\; \Sigma_{\lambda}\, x_j^{\lambda} (x_i^{\lambda})^T - p/N \,.\, I$$

Onde I é matriz identidade.

OBS: a saída de cada neurônio é retornado para todos os neur.

a matriz de pesos é simétrica, i.é., $W = W^T$



LABIC

- Fase de Recuperação: um padrão incompleto x é apresentado à rede e a rede vai tentar recuperar o padrão y que é mais parecido a ele, de acordo com:

$y = \text{sinal}(Wy - \theta)$ - Condição de Alinhamento

onde W é a matriz peso e $\theta$ é o vetor threshold aplicado externamente.



LABIC

## Modelo de Hopfield

- Um modelo de memória associativa usando a regra de Hebb para todos os pares ji possíveis com unidades binárias e atualização assíncrona é chamada um Modelo de Hopfield.
- Em 1982, Hopfield introduziu a idéia de uma função de energia na teoria de RNA:

$H = -0.5\; \Sigma\; w_{ji}\, S_i\, S_j$ é uma função de configuração do sistema.

LABIC

## Modelo de Hopfield

- A propriedade central de uma função Energia é que ela sempre diminui (ou permanece constante) quando o sistema evolui de acordo com sua dinâmica. Assim, os atratores da Figura 1 são os mínimos locais da superfície energia.



LABIC

## Exemplo

- Dada a matriz de conexões:

$$W = 1/3 \begin{bmatrix} 0 & -2 & +2 \\ -2 & 0 & -2 \\ +2 & -2 & 0 \end{bmatrix}$$

Verificar que dos 8 estados possíveis, apenas os estados: (1, -1, 1) e (-1, 1, -1) são estáveis, isto é, satisfazem a condição de alinhamento.



LABIC

## Algoritmo

- Dados M padrões exemplares de compr. N
- Passo1: Conexões (baseada nos modelos exemplares)

$$w_{ji} = \begin{cases} 1/N \sum_{\lambda=0}^{M-1} x_j^{\lambda} x_i^{\lambda} & i \neq j \\ 0 & i = j \quad 0 \leq i,j \leq N-1 \end{cases}$$

- Passo2: **Enquanto** (houver modelos desconh.para serem recuperados) **faça**

1. Acrescentar um novo modelo (padrão de entrada)

   $y_j(0) = x_j \; ; \; 0 \leq j \leq N-1$

2. **Repita**

   $y_j(t+1) = \text{sinal}(\sum_{i=0}^{N-1} w_{ji} y_i(t)), \quad 0 \leq j \leq N-1$

   **até que** $y_j(t+1)$ seja aproximadamente $y_j(t)$.



LABIC

## Modelo de Hopfield

- Limitações
- O no. de padrões M que podem ser armazenados é muito limitado

  $M \leq 0.15 * N$

- Um padrão exemplar será instável se ele tiver muitos bits em comum com outros padrões exemplares. Um modelo é considerado instável se ele é aplicado a rede no tempo zero e a rede converge para um modelo desconhecido.



LABIC

## Modelo de Hopfield

- Estas limitações podem ser eliminadas se os padrões exemplares foram LI. e se possíveis ortogonais.
- Se N=1000, a complexidade de ligações é muito grande:

  $W = (W_{ij})$ ==> 1000 x 1000 conexões



LABIC

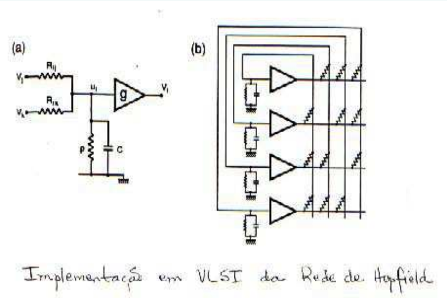
## Modelo de Hopfield

- Existem estudos para se eliminar algumas conexões
- Impor um limiar para existência da conexão
- Se $|w_{ij}| \leq$ valor limiar ==> $w_{ij} = 0$

  caso contrário, permanece constante
- O valor de $t_{ij}$ é no máximo M/N. Os valores variam entre –M/N e M/N.
- $w_{ij}$ = sinal $(w_{ij})$ simplifica-se a ligação(inibitória ou excitatória)

LABIC

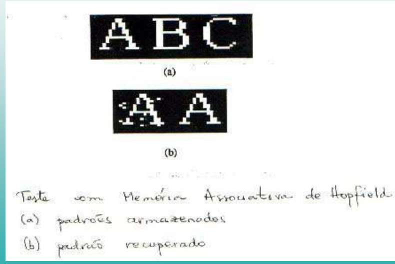
## Rede de Hopfield-circuito






LABIC

## Exemplo de Memória Associativa






LABIC

## Aplicação II - Otimização Combinatorial

- Probl. de Otimização: são problemas cuja solução é um membro de um conjunto de soluções factíveis.
- A solução ótima é uma sol. Factível que minimiza (ou maximiza) uma função custo.
- Exemplo: encontrar o menor caminho entre dois pontos numa rede de comunicação que consiste de centenas de nós, onde cada nó é conectado a um subconj. de outros nós.
- Caixeiro Viajante, Probl. da Mochila, JSS, etc...
- Polinomial, Polinomial Não-determinístico (NP), Exponencial (EXP)



LABIC

## Aplicação II

- Algoritmo tempo polinomial : $O(n^k)$, k=cte, n-tamanho
- Algoritmo Exponencial: rodam em $O(k^n)$ - tempo
- Os probl. NP são considerados difíceis porque o melhor algoritmo para resolvê-lo parece rodar em tempo exp.
- Eles são probl., nos quais, uma boa solução subótima pode ser encontrada em tempo polinomial
- A rede de Hopfield pode ser usada para calcular uma boa solução para estes probl. Num modo eficiente.



LABIC

## Aplicação II

- Probl. Do Caixeiro Viajante

O caixeiro deve visitar n cidades: A, B,C, ... cujas distâncias entre elas são conhecidas.

Objetivo: O probl. é encontrar um caminho fechado que passa por todas as cidades uma única vez, que é MÍNIMO.

- Representação de rotas com saídas da rede

A cidade C é visit. na 6a. Posição: (0 0 0 0 0 1 0 0 0 .......0)

- Se tiver n cidades ==> n vetores com n posições
- Total de Neurônios: $N = n^2$ neurônios



LABIC

## Aplicação II

- $V_{Xi}$ = 0 - a cidade X não é visitada nesta posição
  1 - a cidade X é visitada na i-ésima posição
- Hopfield provou que os pontos de equilíbrio da rede são pontos de mínimo da função de Energia:

$$E = -0.5 \ \Sigma_{i=1}^{N} \Sigma_{j=1}^{N} T_{ij} V_i V_j \ - \Sigma_{i=1}^{N} V_i I_i$$

- $T_{ij}$ = ? $I_j$ = ?
- E = E1 + E2 + E3 + E4

LABIC

# Aplicação II

- E1 : uma cidade deve ser visitada uma única vez
- E2: duas cidades não podem ser visitadas ao mesmo tempo
- E3: As n cidades devem ser visitadas
- E4: comprimento do caminho deve ser mínimo

$E1 = A/2 \, \Sigma_X \Sigma_i \Sigma_{j \; i \neq j} V_{Xi} V_{Xj}$

$E2 = B/2 \, \Sigma_X \Sigma_{Y \neq X} \Sigma_i V_{Xi} V_{Yi}$

$E3 = C/2 \, (\Sigma_X \Sigma_i V_{Xi} - n)^2$

$E4 = D/2 \, \Sigma_i \Sigma_X \Sigma_{Y \neq X} d_{xy} V_{Xi} (V_{Y,,i+1} + V_{Y,,i-1})$



LABIC

- A matriz de conexões tem a seguinte forma:

$$T_{Xi,Yj} = \{ -A\delta_{XY}(1-\delta_{ij}) - B\delta_{ij}(1-\delta_{XY}) - C$$

$$-Dd_{XY}(\delta_{j,i+1} + \delta_{j,i-1})$$

$$onde \, \delta_{ij} = \begin{cases} 1, i = j \\ 0, i \neq j \end{cases}$$

$$I_{xi} = Cn : entradas - externas$$



LABIC


- HOPFIELD, J.J. and TANK, D.W., “Computing with neural networks: A model”, Science, vol. 233, pp. 625-633, aug., 1986.
- HOPFIELD, J.J. and TANK, D.W.,”Neural Computation of Decisions in Optimization Problems”, Biol. Cybern., vol. 2, pp. 141-152, 1985.