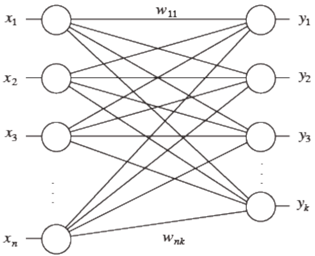
Opakování: Asociativní paměti

Co bylo minule:

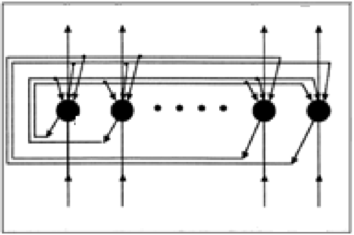
- Asociativní sítě (pamět, AM)
  - bez zpětné vazby, se zpětnou vazbou (rekurentní)
  - Hebbovské učení (1942) a metoda pseudoinverzní matice (1972).
- BAM (Bidirektivní asociativní paměti)
  - synchronní rekurentní asociativní paměť s obousměrnými synapsemi.



Hopfieldova síť

Dnes:

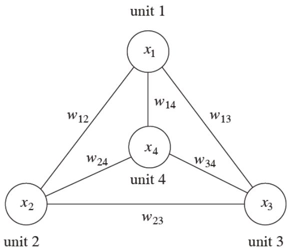
- Hopfieldovy sítě (1982, John Hopfield)
    - Speciální případ asynchronní BAM
      – levá a pravá vrstva neuronů spojeny do jedné



Hopfieldova síť – Architektura

- jen jedna vsrtva s n neurony
- neurony mohou a nemusí mít práh
- skoková přenosová funkce:

$$f(x) = sgn(x) = \begin{cases} 1 & \text{pro } x \geq 0, \\ -1 & \text{pro } x < 0, \end{cases}$$



- biparitní vstupy i výstupy $\{-1; 1\}$
- synaptické hrany s vahami w_{ij} (mezi každýma dvěma neurony i a j)

Hopfieldova síť – Učení a rozpoznávání

Učení s učitelem

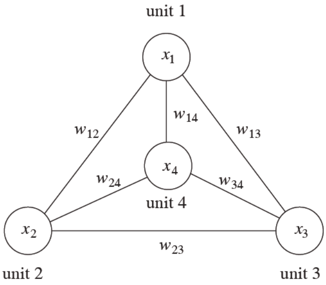
- N trénovacích vzorů (tříd)
- požadované výstupy jsou stejné jako vstupy
- nastavení vah (a prahů)

Rozpoznávání

- asynchronní dynamika

Použití

- Asociativní paměť
- Řešení optimalizačních úloh



Hopfieldova síť - Učení a rozpoznávání

Učení - Hebbovské učení

- Nastavíme hodnoty synaptických vah:

$$w_{ij} = \begin{cases} \sum_{l=1}^{N} x_i^l x_j^l & \text{pro } i \neq j, \\ 0 & \text{pro } i = j, \end{cases}$$

  - $1 \leq i, j \leq n$
  - w_{ij} ... váha hrany (synapse) mezi neuronem i a j
  - $x_i^l \in \{-1; 1\}$ ... i-tá složka l-tého vzoru

- výsledná matice vah:

$$W = \sum_{l=1}^{N} \vec{x}^{lT} \vec{x}^l - NI$$

Hopfieldova síť - Rozpoznávání

- **Inicializace** - předložíme neznámý vstupní vzor $\vec{x}$
  $\rightarrow$ počáteční stav sítě:

$$y_i(0) = x_i \text{ pro } 1 \leq i \leq n$$

  - $x_i \in \{-1; 1\}$ ... i-tá složka předloženého vzoru
  - $y_i(t)$ ... výstup neuronu i v čase t
- **Iterace**

$$y_j(t+1) = f(\sum_{i=1}^{n} w_{ij} y_i(t)) \text{ pro } 1 \leq j \leq n$$

  - f ... (skoková) přenosová funkce

Hopfieldova síť - Rozpoznávání

- **Inicializace** - předložíme neznámý vstupní vzor $\vec{x}$

$$y_i(0) = x_i \text{ pro } 1 \leq i \leq n$$

- **Iterace**

$$y_j(t+1) = f(\sum_{i=1}^{n} w_{ij} y_i(t)) \text{ pro } 1 \leq j \leq n$$

- Iterativní proces se při rozpoznávání opakuje, dokud se výstupy neuronů neustálí.
- Výstupy neuronů pak reprezentují ten trénovací vzor, který nejlépe odpovídá předloženému (neznámému) vzoru.

Hopfieldova síť - Učení a rozpoznávání

Vektor potenciálů po předložení vzoru $\vec{x}^1$:

$$\begin{aligned}
\vec{\xi} &= \vec{x}^1 W = \vec{x}^1 \left(\sum_{l=1}^{N} \vec{x}^{lT} \vec{x}^l - NI\right) \\
&= \vec{x}^1 \vec{x}^{1T} \vec{x}^1 + \vec{x}^1 \vec{x}^{2T} \vec{x}^2 + \ldots + \vec{x}^1 \vec{x}^{NT} \vec{x}^N - N\vec{x}^1 I = \\
&= n\vec{x}^1 + \alpha_{12}\vec{x}^2 + \ldots + \alpha_{1N}\vec{x}^N - N\vec{x}^1 = \\
&= (n - N)\vec{x}^1 + \sum_{l=2}^{N} \alpha_{1l}\vec{x}^l
\end{aligned}$$

- $W = \sum_{l=1}^{N} \vec{x}^{lT} \vec{x}^l - NI$ ... matice vah Hopfieldovy sítě
- α_{1l} ... skalární součin $\vec{x}^1$ a $\vec{x}^l$
- $\sum_{l=2}^{N} \alpha_{1l}\vec{x}^l$ ... perturbační člen (crosstalk)

Hopfieldova síť - Učení a rozpoznávání

Vektor potenciálů po předložení vzoru $\vec{x}^1$:

$$\vec{\xi} \;=\; (n-N)\vec{x}^1 + \sum_{l=2}^{N} \alpha_{1l}\vec{x}^l$$

- Chceme, aby vektor $\vec{x}^1$ byl stabilní stav sítě (tj. $sgn(\vec{\xi}) = sgn(\vec{x}^1)$)
  $\rightarrow$ musí být $N < n$ a perturbační člen $(\sum_{l=2}^{N} \alpha_{1l}\vec{x}^l)$ malý

$\rightarrow$ do paměti lze vložit malý počet navzájem ortogonálních vzorů

- Kapacita $N < 0.15\ n$ nebo $N \sim n/(2logn)$
- Problém se stabilitou $\rightarrow$ ortogonalizace

Hopfieldova síť - Učení a rozpoznávání

Podmínky konvergence ke stabilnímu řešení při rozpoznávání:

- Symetrické váhy $w_{ij} = w_{ji}$
- $w_{ii} = 0$
  $\rightarrow$ Symetrická váhová matice s nulovou diagonálou
- Asynchronní aktualizace výstupu jednotlivých neuronů
  - Stav neuronů zůstává stejný, dokud nejsou vybrány k aktualizaci
  - Výběr pro aktualizaci se provádí náhodně
  - Pravděpodobnost, že budou dva neurony současně měnit svůj stav, je nulová

Příklady

Hopfieldova síť - Učení a rozpoznávání

Věta

- Hopfieldova síť s asynchronní dynamikou dosáhne z libovolného počátečního stavu sítě stabilního stavu v lokálním minimu energetické funkce.

Energetická funkce Hopfieldovy sítě

$$E(\vec{x}) = -\frac{1}{2}\vec{x}W\vec{x}^T = -1/2\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}x_i x_j$$

Pro síť s prahovými neurony:

$$E(\vec{x}) = -\frac{1}{2}\vec{x}W\vec{x}^T + \vec{\theta}\vec{x}^T = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}x_i x_j + \sum_{i=1}^{n}\theta_i x_i$$

Hopfieldova síť - Energetická funkce

Věta

- Hopfieldova síť s asynchronní dynamikou dosáhne z libovolného počátečního stavu sítě stabilního stavu v lokálním minimu energetické funkce.

Náznak důkazu

- Lokální minima energetické funkce odpovídají stabilním stavům.
- Každá změna stavu některého neuronu vede ke snížení celkové energie sítě.
- Stavů sítě je konečně mnoho $\rightarrow$ výpočet skončí v konečném počtu kroků.

Hopfieldova síť - Energetická funkce

Náznak důkazu – pokračování

- Počáteční stav sítě $\sim$ předložený vzor $\vec{x} = \{x_1, \ldots, \underline{x_k}, \ldots, x_n\}$
  Odpovídá mu energie:

$$E(\vec{x}) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}x_ix_j + \sum_{i=1}^{n}\theta_i x_i$$

- K aktualizaci vyberu neuron k
  - k nezmění svůj stav $\rightarrow E(\vec{x})$ se nezmění
  - k změní svůj stav $\rightarrow E(\vec{x})$ se sníží (...)

Hopfieldova síť - Energetická funkce

Náznak důkazu – dokončení

- Nový stav sítě $\sim \vec{x'} = \{x_1, \ldots, x'_k, \ldots, x_n\}$
  Odpovídá mu energie:

$$E(\vec{x'}) = -\frac{1}{2}\sum_{i\neq k}^{n}\sum_{j\neq k}^{n} w_{ij}x_ix_j - \sum_{i=1}^{n} w_{ik}x_ix'_k + \sum_{i\neq k}^{n}\theta_i x_i + \theta_k {x'}_k$$

- Rozdíl energií v původním a novém stavu sítě:

$$\begin{aligned} E(\vec{x}) - E(\vec{x'}) &= -\sum_{i=1}^{n} w_{ik}x_ix_k + \theta_k x_k - \left(-\sum_{i=1}^{n} w_{ik}x_ix'_k + \theta_k {x'}_k\right) \\ &= -(x_k - {x'}_k)\left(\sum_{i=1}^{n} w_{ik}x_i - \theta_k\right) \\ &= -(x_k - {x'}_k)\xi_k > 0 \end{aligned}$$

$\rightarrow$ nový stav má nižší energii

Perceptronové učení pro Hopfieldův model

Motivace

- V některých případech nelze nalézt pomocí Hebbovského učení váhovou matici Hopfieldovy sítě tak, aby N daných vektorů odpovídalo stabilním stavům sítě (i když taková matice existuje)
    - pokud leží vektory, které mají být do sítě uloženy, hodně blízko, může být ,,crosstalk" příliš velký
      $\rightarrow$ horší výsledky Hebbovského učení
- **Alternativa:** perceptronové učení pro Hopfieldovy sítě

Perceptronové učení pro Hopfieldův model

- **Je dáno:** Hopfieldova síť s n neurony, které mají nenulový práh a skokovou přenosovou funkci.
- Má-li si síť ,,zapamatovat" vektor $\vec{x} = (x_1, \dots, x_n)$, bude tento vektor odpovídat stabilnímu stavu sítě tehdy, jestliže se po jeho ,,předložení" stav sítě nezmění
  $\rightarrow$ potenciál každého neuronu by měl mít stejné znaménko jako jeho předchozí stav

Perceptronové učení pro Hopfieldův model

Měly by platit následující nerovnosti:

- pro neuron 1: $sgn(x_1)(0 + x_2 w_{12} + \ldots + x_n w_{1n} - \theta_1) > 0$
- pro neuron 2: $sgn(x_2)(x_1 w_{21} + 0 + \ldots + x_n w_{2n} - \theta_2) > 0$
- ...
- pro neuron n:
  $sgn(x_n)(x_1 w_{n1} + x_2 w_{n2} + \ldots + x_{n-1} w_{n(n-1)} + 0 - \theta_n) > 0$

Protože $w_{ij} = w_{ji} \rightarrow$ nenulových prvků váhové matice je $n(n-1)/2$ a prahů n
$\rightarrow$ vytvoříme vektor $\vec{v}$ dimenze $n + n(n-1)/2$:

$$\vec{v} = (w_{12}, w_{13}, \ldots, w_{1n}, w_{23}, w_{24}, \ldots, w_{2n}, \ldots, w_{n-1n}, -\theta_1, \ldots, -\theta_n)$$

(složky odpovídají prvkům w_{ij} nad diagonálou matice W a n prahům se záporným znaménkem)

Perceptronové učení pro Hopfieldův model

Vektor $\vec{x}$ můžeme transformovat do n pomocných vektorů $\vec{z}_1, ..., \vec{z}_n$ dimenze $n + n(n-1)/2$:

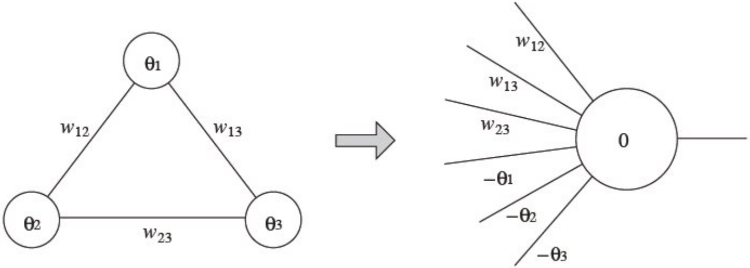
- $\vec{z}_1 = (x_2, x_3, ..., x_n|0, 0, ..., 0|0|1, 0, ..., 0)$
- $\vec{z}_2 = (x_1, 0, ..., 0|x_3, ..., x_n|0, 0, ..., 0|0, 1, ..., 0)$
- ...
- $\vec{z}_n = (0, 0, ..., x_1|0, 0, ..., x_2|0, 0, ...|x_{n-1}|0, 0, ..., 1)$

Potom lze přepsat nerovnosti na:

- pro neuron 1: $sgn(x_1)\vec{z}_1\vec{v} > 0$
- pro neuron 2: $sgn(x_2)\vec{z}_2\vec{v} > 0$
- ...
- Npro neuron n: $sgn(x_n)\vec{z}_n\vec{v} > 0$

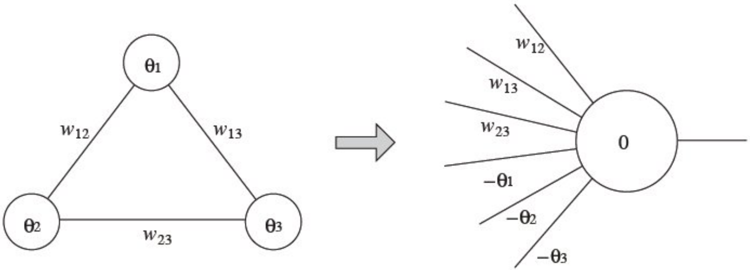
Perceptronové učení pro Hopfieldův model

- k lineární separaci vektorů $\vec{z}_1, \ldots, \vec{z}_n$ podle $sgn(x_i)$ lze použít perceptronové učení $\rightarrow$
  1. pomocí perceptronu spočteme vektor vah $\vec{v}$ nutný pro lineární separaci $\vec{z}_1, \ldots, \vec{z}_n$
  2. nastavíme váhovou matici W a prahy Hopfieldovy sítě podle $\vec{v}$



Perceptronové učení pro Hopfieldův model

- Pokud si má Hopfieldova síť „zapamatovat" N vektorů, je třeba použít popsanou transformaci pro každý vektor $\rightarrow N.n$ pomocných vektorů, které je třeba lineárně odseparovat
- Pokud jsou vektory lineárně separabilní, najde perceptronové učení řešení „zakódované" ve formě $\vec{v}$



Hopfieldova síť pro řešení optimalizačních úloh

- binární kódování
- kódování optializačních problémů pomocí energetické funkce

MULTIFLOP

- $x_1, \dots, x_n$ ... binární stavy jednotlivých neuronů Hopfieldovy sítě
- Síť by se měla dostat do stavu, kdy bude právě 1 neuron aktivní; stav všech ostatních neuronů by měl být 0
- **Cíl:** nalézt minimum funkce $E(x_1, \dots, x_n)$

$$E(x_1, \dots, x_n) = (\sum_{i=1}^{n} x_i - 1)^2$$

Hopfieldova síť pro řešení optimalizačních úloh

MULTIFLOP

$$
\begin{aligned}
E(x_1, \ldots, x_n) &= \left(\sum_{i=1}^{n} x_i - 1\right)^2 = \sum_{i=1}^{n} x_i^2 + \sum_{i \neq j}^{n} x_i x_j - 2\sum_{i=1}^{n} x_i + 1 \\
&= \sum_{i \neq j}^{n} x_i x_j - \sum_{i=1}^{n} x_i + 1 \\
&= -1/2 \sum_{i \neq j}^{n} (-2) x_i x_j + \sum_{i=1}^{n} (-1) x_i + 1
\end{aligned}
$$

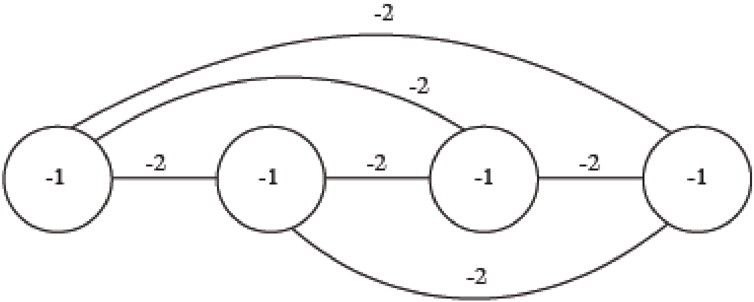
$\rightarrow$ nastavíme váhy a prahy sítě...

Hopfieldova síť pro řešení optimalizačních úloh

MULTIFLOP

$$E(x_1, ..., x_n) = -1/2 \sum_{i \neq j}^{n} (-2) x_i x_j + \sum_{i=1}^{n} (-1) x_i + 1$$

$\rightarrow$ nastavíme váhy a prahy sítě:



Hopfieldova síť pro řešení optimalizačních úloh

Problém n věží

- Umístit na šachovnici $(n \times n)$ n věží tak, aby se navzájem neohrožovaly
  $\rightarrow$ každá věž by měla být v jiném řádku i sloupci než ostatní
  - x_{ij} ... stav neuronu na pozici (i, j) šachovnice $(n \times n)$
  - $\sum_{i=1}^{n} x_{ij}$ ... počet stavů $= 1$ ve sloupci j
  - $\sum_{j=1}^{n} x_{ij}$ ... počet stavů $= 1$ v řádku i
  $\rightarrow$ v každém sloupci a řádku by měla být jen jedna „jednička"

Hopfieldova síť pro řešení optimalizačních úloh

Problém n věží

Cíl: nalézt minimum funkce

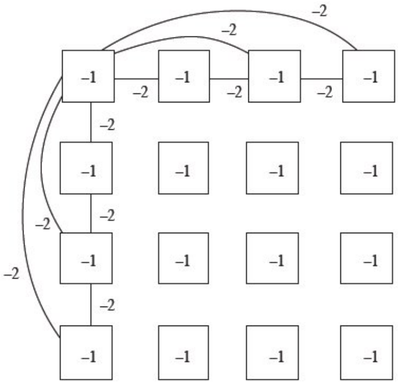
$$\begin{aligned}
E(x_{11},\dots,x_{nn}) &= E_1(x_{11},\dots,x_{nn}) + E_2(x_{11},\dots,x_{nn}) \\
E_1(x_{11},\dots,x_{nn}) &= \sum_{j=1}^{n}(\sum_{i=1}^{n} x_{ij} - 1)^2 \\
E_2(x_{11},\dots,x_{nn}) &= \sum_{i=1}^{n}(\sum_{j=1}^{n} x_{ij} - 1)^2
\end{aligned}$$

... rozšíření multiflopu

Hopfieldova síť pro řešení optimalizačních úloh

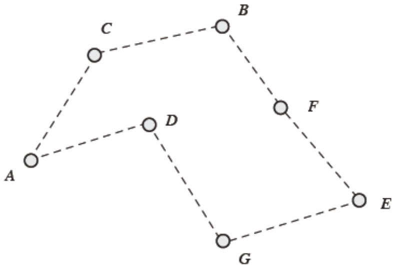
Problém n věží

→ nastavíme váhy a prahy sítě:



Hopfieldova síť pro řešení optimalizačních úloh

TSP (Problém obchodního cestujícího) (NP-úplný problém)



- Nalézt cestu přes n měst $M_1, \ldots, M_n$ tak, aby bylo každé město navštíveno alespoň jednou a délka „okružní jízdy" byla minimální

Hopfieldova síť pro řešení optimalizačních úloh

TSP (Problém obchodního cestujícího)

- Reprezentace pomocí matice, kde řádky odpovídají městům a sloupce odpovídají pořadí návštěvy
  - x_{ik} ... stav neuronu na pozici (i, k) matice
  - $x_{ik} = x_{jk+1} = 1$ ... město M_i bylo navštíveno v k-tém kroce a město M_j bylo navštíveno v $(k+1)$-ním kroce
  - d_{ij} ... vzdálenost mezi městy M_i a M_j
    $\rightarrow$ d_{ij} musíme přičíst k celkové délce cesty

Hopfieldova síť pro řešení optimalizačních úloh

TSP (Problém obchodního cestujícího)

- **Cíl:** minimalizace délky cesty:

$$L = 1/2 \sum_{i,j,k}^{n} d_{ij} x_{ik} x_{jk+1}$$

- **Ale:** povolena jediná návštěva vždy jen jednoho města $\rightarrow$ přidáme požadavky na přípustnou cestu:

$$\begin{aligned} E &= 1/2 \sum_{i,j,k}^{n} d_{ij} x_{ik} x_{jk+1} + \\ &+ \gamma/2 \left[\sum_{j=1}^{n} (\sum_{i=1}^{n} x_{ij} - 1)^2 + \sum_{i=1}^{n} (\sum_{j=1}^{n} x_{ij} - 1)^2) \right] \end{aligned}$$

TSP (Problém obchodního cestujícího)

$$
\begin{aligned}
E &= 1/2 \sum_{i,j,k}^{n} d_{ij} x_{ik} x_{jk+1} + \\
&+ \gamma/2 \left[\sum_{j=1}^{n} (\sum_{i=1}^{n} x_{ij} - 1)^2 + \sum_{i=1}^{n} (\sum_{j=1}^{n} x_{ij} - 1)^2) \right] \\
&= -1/2 \sum_{i,j,k}^{n} (-d_{ij}) x_{ik} x_{jk+1} + \\
&- 1/2 \sum_{i \neq j}^{n} (-\gamma) x_{.i} x_{.j} + \sum_{i=1}^{n} (-\gamma/2) x_{.i} + \gamma/2 + \\
&- 1/2 \sum_{j \neq i}^{n} (-\gamma) x_{i.} x_{j.} + \sum_{j=1}^{n} (-\gamma/2) x_{j.} + \gamma/2
\end{aligned}
$$

Hopfieldova síť pro řešení optimalizačních úloh

TSP (Problém obchodního cestujícího)

$\rightarrow$ nastavení vah a prahů sítě:

- $w_{ik,jk+1} = -d_{ij} + t_{ik,jk+1}$, kde

$$t_{ik,jk+1} = \begin{cases} -\gamma & \text{pro neurony ve stejném řádku nebo sloupci,} \\ 0 & \text{jinak} \end{cases}$$

- $\theta_{ij} = -\gamma/2$

Stochastické modely neuronových sítí

- Hopfieldův model se používá k řešení optimalizačních problémů, které lze vyjádřit ve formě minimalizované energetické funkce (i když není zaručeno nalezení globálního optima)
- **Problém:** zabránit ,,uvíznutí" v lokálním minimu energetické funkce

Stochastické modely neuronových sítí

Modifikace Hopfieldova modelu k zlepšení konvergence a omezení lokálních minim

1. strategie: zvětšení počtu možných cest k řešení ve stavovém prostoru
   → dovolit i stavy ve formě reálných hodnot (sigmoidální přenosová funkce)
   → **spojitý model**
2. strategie: omezení lokálních minim energetické funkce pomocí „zašuměné dynamiky sítě"
   → dočasné povolení aktualizace stavu sítě i za cenu přechodného zvýšení energetické hladiny
   → **stochastické modely: simulované žíhání, Boltzmannův stroj**

Spojitý model Hopfieldovy sítě

- Aktivace neuonu i vybraného pro aktualizaci podle:

$$x_i = s(\xi_i) = \frac{1}{1 + e^{-\xi_i}}$$

  ξ_i je potenciál (excitace) neuronu i

- Pomalá změna potenciálu neuronu v čase podle:

$$\xi_i(t+1) = (1-\alpha)\xi_i(t) + \alpha \sum_{j=1}^{n} w_{ij} x_j = (1-\alpha)\xi_i(t) + \alpha \sum_{j=1}^{n} w_{ij} s(\xi_j)$$

- $0 \leq \alpha \leq 1$ ... adaptační parametr
- w_{ij} ... váha mezi neuronem i a j

Spojitý model Hopfieldovy sítě

- Tedy:

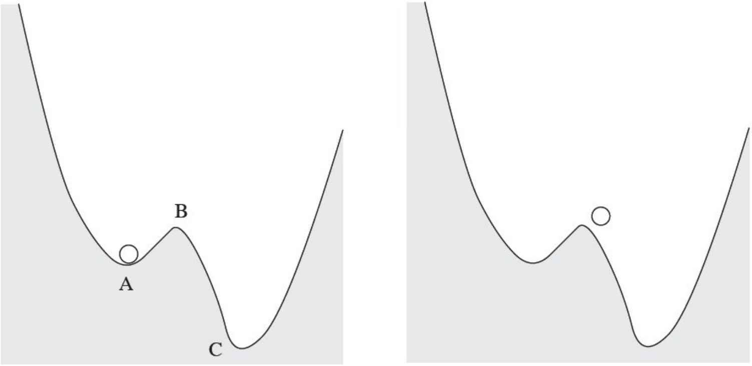
$$\Delta\xi_i(t) = -\alpha\xi_i(t) + \alpha \sum_{j=1}^{n} w_{ij} s(\xi_j)$$

- Při simulacích spočítat diskrétní aproximaci $\Delta\xi_i(t)$ a přičíst ji k aktuální hodnotě $\xi_i(t)$
  výsledkem bude nový stav $x_i = s(\xi_i)$
- Asynchronní dynamika – vede k dosažení rovnovážného stavu

Spojitý model Hopfieldovy sítě

- Pro kombinatorické problémy může spojitý model nalézt lepší řešení než model diskrétní
- Pro velmi složité problémy (typu TSP) ovšem spojitý model obecně nenachází výrazně lepší řešení
- Snadná hardwarová implementace

Stochastické systémy - simulované žíhání



Simulované žíhání

Při minimalizaci energetické funkce E se tento jev simuluje následujícím způsobem:

- Hodnota proměnné x se změní vždy, když může aktualizace Δx zmenšit hodnotu energetické funkce E
- Pokud by se při aktualizaci x naopak hodnota E zvýšila o ΔE, bude nová hodnota x (tj. $x + \Delta x$) přijata s pravděpodobností $p_{\Delta E}$:

  $$p_{\Delta E} = \frac{1}{1 + e^{\Delta E / T}}$$

  kde T je tzv. teplotní konstanta.

Simulované žíhání

- Pro velké hodnoty T bude $p_{\Delta E} \sim 1/2$ a aktualizace stavu nastane zhruba v polovině těchto případů
- Pro T $= 0$ bude docházet pouze k takovým aktualizacím, kdy se hodnota E sníží
- Postupná změna hodnot T z velmi vysokých hodnot směrem k nule odpovídá zahřátí a postupnému ochlazování v procesu žíhání

Navíc:

- Lze ukázat, že touto strategií lze dosáhnout (asymptoticky) globálního minima energetické funkce
- Sigmoida nejlépe odpovídá funkcím používaným v termodynamice (pro analýzu teplotní rovnováhy)

Stochastické systémy - Boltzmannův stroj

Definice

- Boltzmannův stroj je Hopfieldova síť, která se skládá z n neuronů se stavy $x_1, x_2, \ldots, x_n$.
  Stav neuronu i se aktualizuje asynchronně podle pravidla:

$$x_i = \begin{cases} 1 & \text{s pravděpodobností } p_i, \\ 0 & \text{s pravděpodobností } 1 - p_i, \end{cases}$$

  kde

$$p_i = \frac{1}{1 + e^{-(\sum_{j=1}^{n} w_{ij} x_j - \theta_i)/T}}$$

- T je kladná teplotní konstanta
- w_{ij} a θ_i jsou váhy a prahy neuronů

Boltzmannův stroj

Energetická funkce Boltzmannova stroje

$$E(\vec{x}) = -1/2\vec{x}W\vec{x}^T + \vec{\theta}\vec{x}^T = -1/2\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}x_ix_j + \sum_{i=1}^{n}\theta_i x_i$$

- Rozdíl mezi Boltzmannovým strojem a Hopfieldovým modelem spočíváve stochastické aktivaci neuronů

Boltzmannův stroj

- Pokud je T velmi malé, bude:
  $p_i \sim 1$, jestliže je excitace neuronu kladná

$$\sum_{j=1}^{n} w_{ij} x_j - \theta_i > 0$$

  $p_i \sim 0$, jestliže je excitace neuronu záporná
  $\rightarrow$ dynamika Boltzmannova stroje aproximuje dynamiku diskrétní Hopfieldovy sítě a Boltzmannův stroj najde lokální minimum energetické funkce

Boltzmannův stroj

- Pro $T > 0$ je pravděpodobnost změny anebo posloupnosti změn ze stavu $x_1, x_2, \ldots, x_n$ do jiného stavu vždy nenulová.
  $\rightarrow$ Boltzmannův stroj nezůstane v jediném stavu
  $\rightarrow$ snižování a zároveň možnost zvyšování energie systému
- Pro velké hodnoty T projde síť téměř celý stavový prostor
- V ochlazovací fázi má síť tendenci zůstávat déle v oblastech blízkých atraktorům lokálních minim

Důležité

Pokud se teplota snižuje správným způsobem, můžeme očekávat, že systém dosáhne globálního minima s pravděpodobností 1

Jak je to v Matlabu

- speciální typ sítě – lineárně saturovaná přenosová funkce, synchronní dynamika, matice nemusí být symetrická s nulovou diagonálou
- *newhop* ... vytvoření ,,Hopfieldovy" sítě
  - net = newhop(T)
  - net.LW{1,1} ... matice vah
  - net.b{1,1} ... prahy neuronů
- *sim* ... rozpoznávání
  - [Y,Pf,Af] = sim(net,Pocet_vzoru,[],T).
    Y = Af ... výsledek
  - [Y,Pf,Af] = sim(net,{Pocet_vzoru,Pocet_iteraci},{},T).
    $Af\{1,1\}$ ... výsledek
    $Y\{1,k\}$ ... výsledek po k iteracích
- dema