

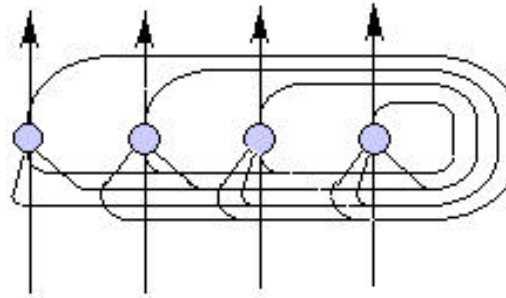
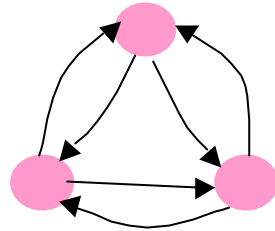
## ■ Hátracsatolt mesterséges neuronháló modellek

A hátracsatolt (feedback) modelleknél az input réteg neuronjai felől nemcsak előre, az output réteg neuronjai felé, hanem visszafelé is. Így egy neuron kimenete visszacsatolódik egy megelőző réteg neuronjára, ezáltal a visszacsatolt neuron bemenetének értéke függhet a korábbi kimeneti értéktől. Ezen hálók asszociatív memóriaként való működésük során az input ráadásakor egy iteratív konvergens folyamaton keresztül veszik fel a stabil állapotukat és szolgáltatják a kimenetet.

Vizsgált típusok:

1. **Hopfield** háló
2. **BAM** = Bidirectional Associative Memory, kétirányú asszociatív memória
3. **ART** = Adaptive Resonance Theory, adaptív rezonancia elmélet alapú háló.

## ■ Hopfield háló, 1982



1.

John Hopfield



Jellemzői:

- Egyetlen neuronréteg, mely bemenetként és kimenetként egyaránt szerepel
- Teljes kapcsolódás: minden neuron minden neuronnal; egyetlen laterális hálózat
- A neuronok kapcsolata szimmetrikus:  $w_{ij} = w_{ji}$
- A kimenetek bemenetekre való visszacsatolása miatt időfüggő működést mutat
- CAM (Content Addressable Memory)-ként, tartalom által címezhető memóriaként működik: képes részinformáció alapján a teljes tárolt minta felidézésére, autoasszociatív módon
- A neuronok küszöbvel rendelkező bináris (0;1), vagy az egyszerűbb matematikára vezető bipoláris (-1;+1) lépcsős függvényrel működnek.
- Az input is kétértékű: bináris, vagy bipoláris.

1. <http://images.google.com/imgres?imgurl=www.ai.univ-paris8.fr/CSAR/images/>

## ■ Hopfield háló ..



Jellemzői:

- A **súlymátrix képzése** nagyon egyszerű:
  - képezni kell egy adott inputvektor mátrixot eredményező **külso szorzatát**
  - a különféle inputvektorokkal adódó **mátrixok összege** lesz a súlymátrix.
- **Működés:** a megkonstruált ("betanított") mátrixnak megadva egy input vektort (mely eltérhet a konstruálásra felhasználtaktól) a háló a visszacsatolt gerjesztések alapján **eljut egy stabil állapotba** (energiaminimum), mely megfelel az egyik korábbi inputnak. 3D-ben az energiafelület egy völgyekkel tarkított felület lesz. Annyi lesz a völgyek száma, ahány korábbi betanító vektort használtunk.
- Egy soha nem látott inputminta hatására a hozzá legközelebb álló lokális **energiaminimum-völgybe** jut el a rendszer, mely megfelel egy olyan tanult mintázatnak, amely legközelebb áll a megadott input mintához. Ezen működési forma miatt nevezik CAM-nek.
- A neuronok **kollektív döntést** hoznak, mivel mindegyik neuron kimenete hat a többi bemenetére.
- N flip-flop: N állapot közül billen be valamelyikbe.

## ■ A Hopfield háló algoritmus



- A kapcsolatok súlyainak megadása (**külső szorzattal** kapott mátrixok összege, a főátlóban 0-ák)

$$w_{ij} = \begin{cases} \sum_{s=0}^{M-1} x_i^s x_j^s & i \neq j \\ 0 & i = j, 0 \leq i, j \leq M-1 \end{cases}$$

Ahol  $w_{ij}$  az  $u_i$  és  $u_j$  neuronok közötti kapcsolat súlya, valamint  $x_i^s$  az  $i$ . eleme az  $s$  betanító mintának, értéke  $-1$ , vagy  $+1$ . A betanítóminták száma  $M$ ,  $0$ -tól  $M-1$ -ig. A neuronok küszöbértéke  $0$ .

- A háló alkalmazása: egy ismeretlen **minta bemutatása**:

$$\mathbf{m}_i(0) = x_i \quad 0 \leq i \leq N-1$$

Ahol  $\mathbf{m}_i(t)$  az  $i$ . neuron kimenete a  $t$ . időciklusban.

- **Iterálás** a konvergencia eléréséig

$$\mathbf{m}_i(t+1) = H\left(\sum_{j=0}^{N-1} w_{ij} \mathbf{m}_j(t)\right) \quad 0 \leq j \leq N-1$$

Az iteráció addig ismétlődő, mígnem a neuronok kimenetei már nem változnak. A  $H()$  a lépcsős függvény.

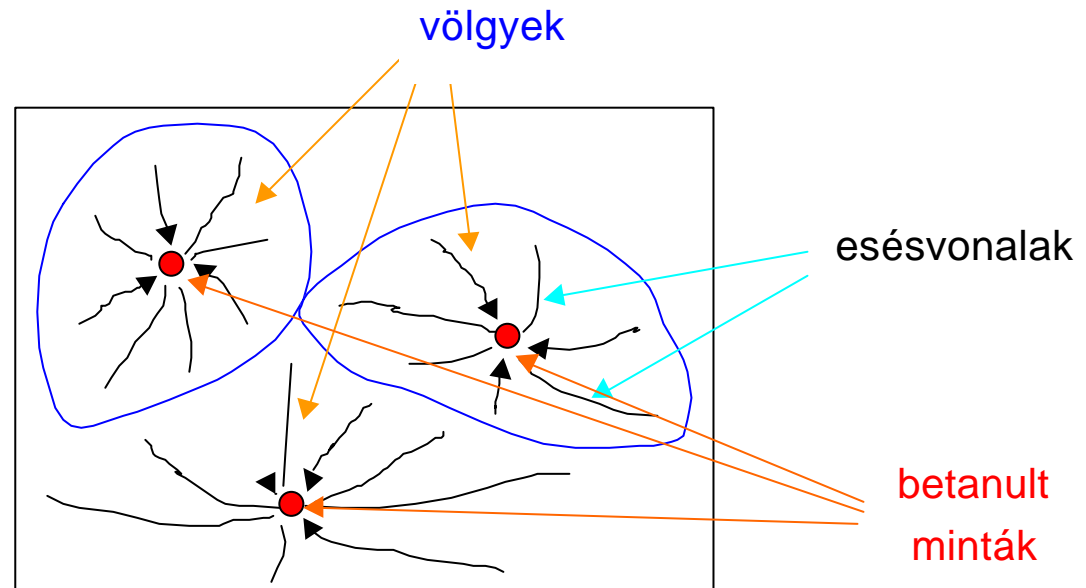
## ■ A Hopfield háló algoritmus ..



Magyarázat:

- Az 1. lépés a háló „betanítása”, amikor is a bemeneti mintához önmagát társítjuk, mint kimeneti mintát.
- A 2. és 3. felismerési fázisban egy bemutatott ismeretlen mintát, mely leggyakrabban valamelyik betanítóminta zajjal szennyezett, vagy hiányos képe kell a hálónak felismernie, azaz a zaj nélküli, teljes mintát kell visszaadnia a beálló stabil állapotban.

## ■ Az energiafelület



Az energiafelülettel lehet szemléltetni a háló működését.

## ■ Az energiafelület



Az energiafelület egyenlete: 
$$E = -\frac{1}{2} \sum_i \sum_{j \neq i} w_{ij} x_i x_j + \sum_i x_i \Theta_i$$

$\Theta_i$  az i. neuron küszöbértéke.

Ha sikerül elérni, hogy a betanítóminták az energiafelület minimumpontjait foglalják el, akkor egy gradiens irányú mozgással minden ismeretlen input mintához előhívhatjuk valamelyik tárolt mintát.

## ■ A betanítóminták eltárolása

Cél: az aktuális betanítóminta számára egy újabb völgy és minimumpont létrehozása a korábban eltárolt minták tárolóvölgyének elrontása nélkül.

Hogy egy eltárolandó új minta esetében az energiaérték ne nőhessen, az első tag mellett a  $\sum_i x_i \Theta_i$  tagnak is nempozitívnak kell lennie. Egy adott betanítóminta esetén a küszöbértékek  $x_i$ -vel ellentett előjelűre választásával ez elérhető, de mivel több eltérő betanítóminta van, emiatt csak a  $\Theta_i = 0$  érték választása ad előjelfüggetlen megoldást ezen tag hatásának eltüntetésére.

## ■ A betanítóminták eltárolása ..



A megmaradt első tag negativitásához vegyük figyelembe, hogy az éppen betanítandó  $s$  minta  $x_i^s$  eleméhez kapcsolódó súlyok a  $w_{ij}^s$  jelűek, ezek megváltoztatására törekszünk. Az összes többi súly, mely a korábbi betanításokra vonatkozó információt hordoz, jó lenne, ha változatlan maradna. Válasszuk ezért két részre az energiafüggvény megmaradt első tagját:

$$E = -\frac{1}{2} \sum_i \sum_{j \neq i} w'_{ij} x_i x_j - \frac{1}{2} \sum_i \sum_{j \neq i} w_{ij}^s x_i^s x_j^s = E_{kivéve s} + E_s$$

Az  $s$  minta eltárolása azt jelenti, hogy a hozzá tartozó energiarészt minél negatívabbra választjuk, előidézve így az energiafelület bemélyülését. Ehhez a

$$\sum_i \sum_{j \neq i} w_{ij}^s x_i^s x_j^s$$

rész minél nagyobbra választásán vezet az út. Amennyiben a  $w_{ij}^s$  súlyt egyenlőnek választjuk az  $x_i^s x_j^s$  szorzattal, úgy elérjük, hogy az  $(x_i^s x_j^s)^2$  kifejezés mindig nemnegatív lesz.

Az összes betanítóminta esetére összegezzük az  $s$  mintákra a fenti kifejezést, hogy megkapjuk a  $w_{ij}$  súlyt az összes minta bemutatása esetére:

$$w_{ij} = \sum_s w_{ij}^s = \sum_s x_i^s x_j^s$$

## ■ A betanítóminták eltárolása ..



Ha összevetjük ezt az eredményt a Hopfield háló algoritmusának első lépésével, akkor azt látjuk, hogy azonos, azaz az első lépés létrehozza a betanítómintáknak megfelelő mélyedéseket az energiafelületen.

## ■ A minták előhívása – az asszociatív memória működése

Egy tetszőleges mintához legközelebb eső eltárolt minta előhívása szemléltethető a tetszőleges mintának megfelelő pontból a legközelebbi völgy aljára való leereszkedéssel, mely kivitelezhető egy gradiens módszerrel. A cél eléréséhez egyetlen  $k$ . kiválasztott neuron energiacsökkenést eredményező változtatását vizsgáljuk, majd ciklusban addig vesszük az egyes neuronokat, amíg az energia csökkenthető. Ehhez alakítsuk át az energiafüggvényt úgy, hogy a  $k$ . neuront kiemeljük:

$$E = -\frac{1}{2} \sum_{i \neq k} \sum_{j \neq k} w_{ij} x_i x_j + \sum_{i \neq k} x_i \Theta_i - \frac{1}{2} x_k \sum_j x_j w_{kj} - \frac{1}{2} x_k \sum_i x_i w_{ik} + x_k \Theta_k$$

Tegyük fel, hogy a  $k$ . neuron állapotát  $x_{k1}$ -ről  $x_{k2}$  értékre változtatja. Az energiafüggvény  $\Delta E$  változását megkapjuk, ha az  $x_{k2}$ , majd az  $x_{k1}$  helyettesítésével kapott energiakifejezéseket kivonjuk.



## ■ A minták előhívása – az asszociatív memória működése..



Az energiaváltozás  $\Delta x_k = x_{k2} - x_{k1}$  helyettesítése után:

$$\Delta E = -\frac{1}{2} \left[ \Delta x_k \sum_j x_j w_{kj} + \Delta x_k \sum_i x_i w_{ik} \right] + \Delta x_k \Theta_k$$

Mivel a súlymátrix szimmetrikus, az indexek felcserélhetők és kapjuk:

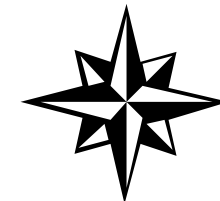
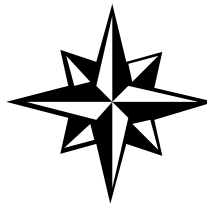
$$\Delta E = -\Delta x_k \left[ \sum_j x_j w_{kj} - \Theta_k \right]$$

A zárójelbeli rész a k. neuron aktivációs potenciálja, a  $\Theta_k$  küszöbértéket pedig nullára választottuk. Ahhoz, hogy az energia ne nőjön, hanem inkább csökkenjen, pozitív aktiváció esetén pozitív  $\Delta x_k$  változás kell, negatív aktiváció esetén pedig negatív  $\Delta x_k$  változás. Az algoritmus 3. lépésében adott H() függvénnyel származtatott iteráló függvényérték megvalósítja ezt az elvárást. Ez azt jelenti, hogy egymást követő egyre alacsonyabb energiájú állapotokon keresztül a háló eljut egy lokális minimumba, az ott eltárolt mintát kiadva stabil kimenetként.

## ■ A minták előhívása – az asszociatív memória működése..



- A konvergencia a fenti **aszinkron** módszeren túl **szinkron** módszerrel is végezhető, amikor is az összes neuronra elvégezzük a kimenet számítását a megelőző időciklusban számított kimeneteket a bemenetekre ráadva. Az aszinkron eljárás véletlenszerűen választja ki a számításhoz a neuront, ezért a folyamat nem lesz egyértelmű. Kimutatták, hogy nincs jelentős eltérés, bár csak az aszinkron bizonyítottan konvergens, szinkron esetben egy átmeneti szakasz után előfordulhat, hogy két minta váltakozik a végtelenségig.
- A használható működéshez a betanítóminták száma nem lehet több a neuronok **15 százalékánál**. Ellenkező esetben a betanított mintáknak meg nem felelő output (völgy) is adódhat (**metastabil** állapot).
- Léteznek a fentitől eltérő **folytonos idejű** Hopfield hálózatok is, melyek vonzó tulajdonsága az **analóg hardveren** való implementálhatóság.

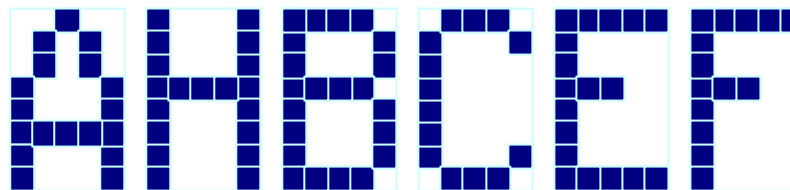


## ■ Példa a Hopfield háló alkalmazására

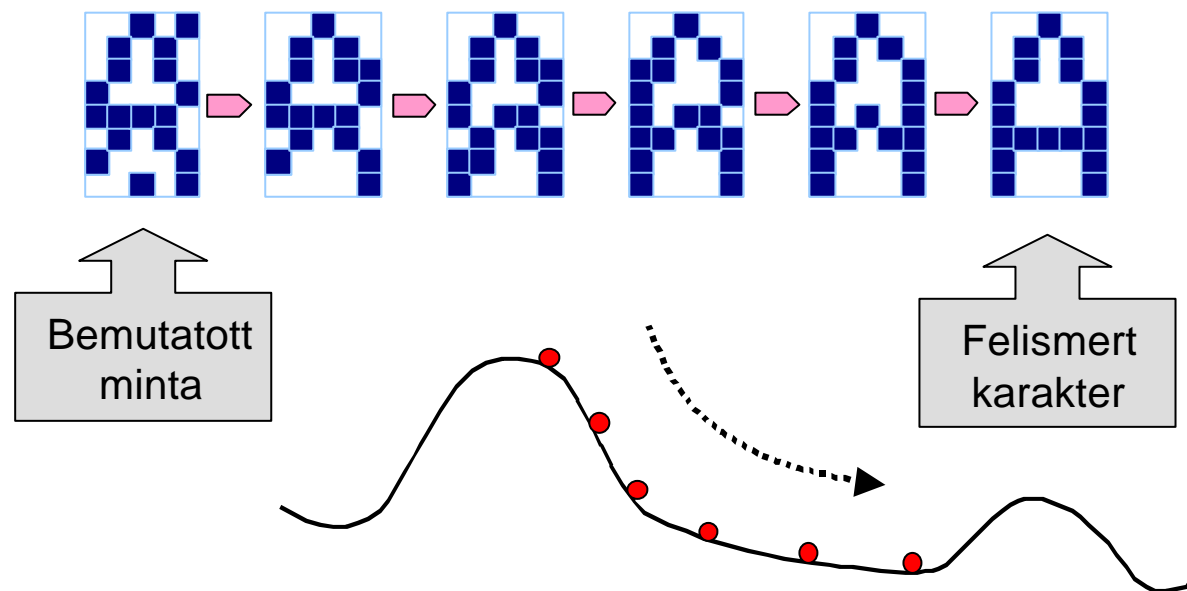


- **Karakterfelismerő**

Adottak a betanítóminta-karakterek:



- A bemutatott minta és az állapotváltozások a stabil állapotig:



## ■ A metastabil állapotok kiküszöbölése: a Boltzmann gép



A Hopfield háló olyan lokális magaslati gödrökben is elakadhat, melyek nem tartoznak a betanított mintákhoz. Ha ki akarunk kerülni ezekből, hogy a mélyebben lévő valódi völgyek aljára jussunk, a Boltzmann gép alkalmazható. A Boltzmann gép a **szimulált hűtés modellt** alkalmazza, mely úgy is interpretálható, hogy a völgy felé guruló, de egy magaslati gödörben elakadó golyó mozgásához rezgést, „hőmozgást” adunk, miáltal átkerülhet a gödrök peremén. Ennek a járulékos mozgásnak a kiterjedtségét, hatását fokozatosan csökkentjük a keresés előrehaladtával. Végül gradiens kereséshez jutunk.

Minden egyes neuron számít egy energia különbséget:

$$\Delta E_k = \sum_i w_{ki} s_i - \Theta_k$$

és egy  $p_k$  valószínűséggel kerül ebbe az állapotba, ahol a  $p_k$  valószínűség az energiakülönbség és a  $T$  hőmérséklet függvénye (v.ö.: szigmoid!):

$$p_k = \frac{1}{1 + e^{-\Delta E_k / T}}$$

## ■ A metastabil állapotok kiküszöbölése: a Boltzmann gép ..



A háló a nagyszámú globális energiaállapot egyikébe kerülhet, melyek eloszlását a Boltzmann eloszlás adja meg. Ha  $P_a$  annak a valószínűsége, hogy a  $T$  hőmérsékletnél az  $E_a$  energiájú globális állapotba kerül, akkor ezt a Boltzmann eloszlás a következőképpen számítja:

$$P_a = k e^{-E_a / T}$$

Hasonlóan, ha egy  $P_b$  valószínűséggel kerülhet az  $E_b$  energiaszintu helyre, képezhető a következő hányados:

$$\frac{P_a}{P_b} = \frac{e^{-E_a / T}}{e^{-E_b / T}} = e^{-(E_a - E_b) / T}$$

A háló akkor kerülhet termális egyensúlyba, korrekt alacsony energiájú helyre, amikor az állapotokba kerülések valószínűségei már nem változnak, mely dolog az energiaszintektől is függ. Ha az  $E_a$  alacsonyabb energia, mint  $E_b$ , a következőket írhatjuk:

$$E_a < E_b$$
$$e^{-(E_a - E_b) / T} > 1$$

ezért

$$P_a / P_b > 1$$

így

$$P_a > P_b$$

## ■ A metastabil állapotok kiküszöbölése: a Boltzmann gép ..



Ez azt jelenti, hogy amint a háló eléri a hőegyensúlyt, az alacsonyabb energiaszintű állapotok valószínűbbek, és ez csak a relatív energiájuktól függ.

Magasabb hőmérsékleten a hőegyensúly gyorsabban beáll, de jó globális energiaállapotok nem valószínűbbek a rosszaknál. A háló működése közben végzett hőmérsékletcsökkentést szimulált lehűtésnek nevezik, ez lehetővé teszi, hogy a háló gyorsabban elérje az alacsonyabb hőmérsékleti egyensúlyt.

A Boltzmann gép alapú hálóak 0-1 közötti kimenőértéku **szigmoid** átviteli függvény használnak, melynek **meredekségével** lehet változtatni a  **$T$**  hőmérsékletet: kis meredekség felel meg a nagyobb hőmérsékletnek. A szigmoid függvény követi a Boltzmann eloszlást.

## ■ A Boltzmann gép tanulási folyamata



Két fázisú:

Bár a háló **teljes kapcsolatiságú**, de nem minden neuronnak egyforma a szerepe. Az első fázisban el kell dönteni, mely neuronok lesznek a **bemenetiek**, melyek alkotják a **rejtett** réteget és melyek a **kimenetet**.

Az **első fázisban** ráadjuk az **inputot** és az **outputot is**. Ezután egy szimulált hűtési folyamatban a rejtett neuronok eléri a hoegyensúlyt. A gerjesztett, kapcsolódó neuronok kapcsolatai **megerősödnek**.

A **második fázisban csak az input** van ráadva. A háló az előzőekhez hasonlóan fut a hoegyensúlyig, majd a gerjesztett kapcsolódó neuronok kapcsolata **csökkentődik**.

Amíg az első fázis a bemenetet a kimenettel összekötő kapcsolatokat erősíti, addig a második fázis leépíti a gyenge kapcsolatokat.

A Boltzmann gép a legmélyebb globális minimumot rendszerint eléri, mivel képes kikerülni a lokális gödrökből a homozgásnak köszönhető nagyobb energiaszintekre történő ugrások révén.

A lokális minimumokban való **megállapodás valószínűsége** a minimumok által képviselt **energiaszint függvénye**.

## ■ A Boltzmann gép tanulási algoritmus



### Első fázis: megerősítés

1. Input és output ráadása
2. Engedjük a hálót működni. Számítsuk az állapot energiáját

$$\Delta E_k = \sum_i w_{ki} s_i - \Theta_k \quad 0 \leq i \leq N-1$$

Majd lépünk alacsonyabb energiaszintű állapotba  $p_k$  valószínűséggel

$$p_k = \frac{1}{1 + e^{-\Delta E_k / T}}$$

Csökkentsük a  $T$  hőmérsékletet a kimenet stabilá válásáig.

3. Erosítsuk meg a kapcsolati súlyokat azon neuronok között, melyek mindketten tüzelnek.

### Második fázis: gyengítés

1. Csak az inputot adjuk rá
2. Engedjük a hálót a hoegyensúlyt elérni
3. Csökkentsük a súlyokat azon kapcsolatoknál, melyek tüzelő neuronokat kötnek össze.

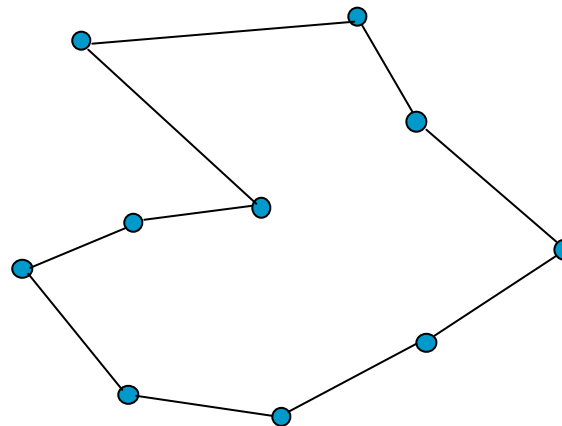


## ■ A Boltzmann gép alkalmazása

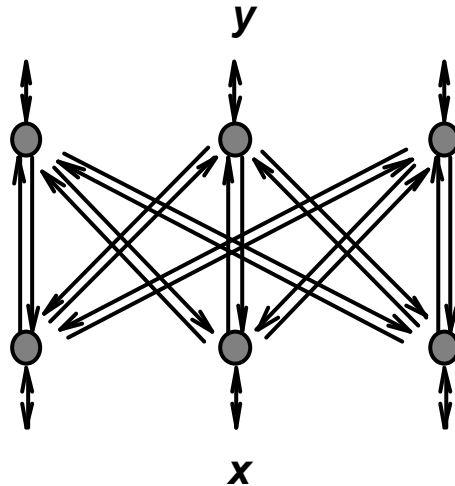


Számtalan, valószínűleg ellentmondó korlátozást tartalmazó korlátozás-kielégítéssel feladatok megoldására is bevált.

- Egyik sikeres alkalmazása az utazó ügynök feladat: legrövidebb körút megtalálása.



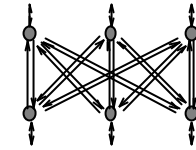
- **BAM** = Bidirectional Associative Memory, kétirányú asszociatív memória, Bart Kosko, 1986



*A Hopfield-háló általánosításának tekinthető szerkezet feltalálója Bart Kosko.*

### Jellemzői

- Két rétegből áll, mindkettő működhet inputként, vagy outputként
- A nevét onnan kapta, hogy a rétegek között kétirányú a kapcsolat, előre- és hátracsatolt vonalak is vannak.
- A súlymátrix konstruált: input-output párokat kell mutatni a hálónak. Ezeket társítja (association).
- Autoasszociatív és heteroasszociatív működésre egyaránt képes



## ■ BAM ..

A súlymátrix előállítása („betanítás”)

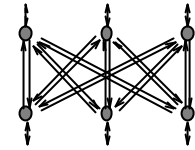
A súlymátrix az input mintapárok mátrixot eredményező külső szorzatával adódó mátrixok összegzésével adódik:

$$\mathbf{W} = \mathbf{y}_1 \mathbf{x}_1^t + \mathbf{y}_2 \mathbf{x}_2^t + \mathbf{y}_s \mathbf{x}_s^t + \dots + \mathbf{y}_M \mathbf{x}_M^t$$

Ahol  $M$  a tárolandó minták száma,  $\mathbf{x}_s$  és  $\mathbf{y}_s$  az  $s$ . mintapár. A kapcsolatok az  $x$  rétegtől az  $y$  felé indexeltek, azaz a  $w_{23}$  súly az  $x$  réteg 3. neuronjától az  $y$  réteg 2. neuronja felé mutat.

- Ha az  $\mathbf{y}$  vektor helyett is az  $\mathbf{x}$  vektort használjuk, autoasszociatív memóriát készíthetünk.

## ■ BAM ..

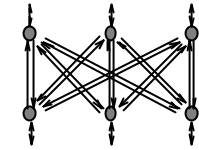


### A háló működése

Ha a súlymátrixot megkonstruáltuk, a háló kész az információ előhívására.

### Az információelőhívás lépései

- Az  $\mathbf{x}_0$ ,  $\mathbf{y}_0$  részleges inputok megadása a bemeneteken.
- Az információ áterjesztése az x rétegről az y rétegre és az y réteg neuronjaira a számítás elvégzése.
- A kiszámított y rétegbeli kimenetek visszaterjesztése az x rétegre és a számítások elvégzése az x réteg neuronjaira.
- A 2. és a 3. lépés ismétlése, mígnem már nincs változás a rétegek kimenő értékeiben. Ekkor az előhívott  $\mathbf{x}$ ,  $\mathbf{y}$  betanított mintapár leolvasható.
- Megjegyzés: a ciklus kezdhető az y rétegről való áterjesztéssel is.
- Egy lehetséges használat: az  $\mathbf{x}_0$  input az egyik korábbi  $\mathbf{x}_s$  betanítómintához hasonlít,  $\mathbf{y}_0$  viszonyáról a betanított  $\mathbf{y}_s$  mintákhoz nem tudunk semmit, nem is adjuk meg. Ekkor a kimenet az  $\mathbf{x}_0$ -hoz közelálló  $\mathbf{x}_s$  és a párja,  $\mathbf{y}_s$  lesz.



## ■ BAM ..

A súlyokkal szorzott bemenetek összegeként előálló aktivációs potenciálból a kimenetek a következőképpen adódnak:

$$y_i(t+1) = \begin{cases} +1 & net_i^y > 0 \\ y_i(t) & net_i^y = 0 \\ -1 & net_i^y < 0 \end{cases}$$

Hasonlóan számíthatók az x réteg kimenetei.

- A BAM energiafüggvénye

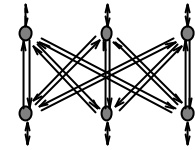
$$E = -\sum_{i=1}^m \sum_{j=1}^n y_i w_{ij} x_j$$

- A háló tárolási **kapacitása**
- *Kosko* szerint a tárolható mintapárok száma az  $m$ ,  $n$  neuronszámok minimuma. *Tanaka* és társai szerint egy  $m=n$  háló max  $1998n$  mintapár tárolására képes.

## ■ BAM ..

- A bemutatott BAM típus a **diszkrét** BAM. Emellett létezik szigmoid, vagy tangens hiperbolikus átviteli függvényt használó **folytonos** BAM is.
- Az interneten kipróbálható a hálók működése.

Hopfield és BAM háló futtatására alkalmas applet:  
<http://www.comp.nus.edu.sg/~pris/AssociativeMemory/DemoApplet.html>



Model

☐ Hopfield Model ☒ Bidirectional Associative Memory Model

Layer 0

Number of Rows 20

Number of Columns 20

Layer 1 (for BAM model only)

Number of Rows 20

Number of Columns 20

New Memory

Learn Pattern

Recall Pattern

Stop Recall

View Pattern

Add Noise

Clear Memory

Clear Maps

Close Memory

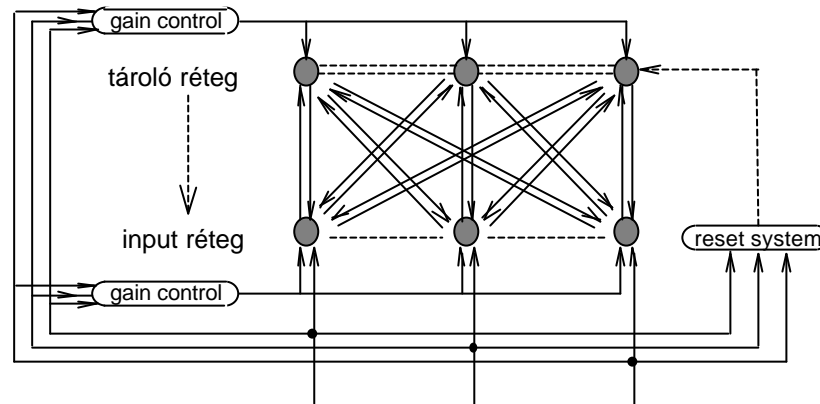
Number of Patterns 0

Pattern Number

Updating Scheme

☒ Synchronous ☐ Asynchronous

- Az ART= Adaptive Resonance Theory, Adaptív Rezonancia Elmélet alapú háló, Stephen Grossberg és Gail Carpenter, 1983



gain control = a nyerést vezérlő rendszer  
reset system = törlő rendszer



### Jellemzői

- Két réteg neuront tartalmaz: 1. input/összehasonlító réteg  
2. output/felismerő/tároló réteg
- Teljes oda-vissza kapcsolat az 1. és 2. réteg között:  $\mathbf{W}$  1.  $\Rightarrow$  2.  
 $\mathbf{T}$  2.  $\Rightarrow$  1.
- A 2. rétegben versenyzés, tiltó laterális kapcsolatok vannak+saját erősítése
- A neuronok valójában neuron csoportok, klusterek

## ■ Adaptív Rezonancia Elmélet alapú háló

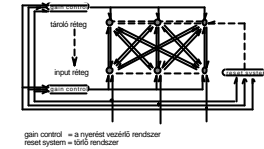
### Működése

#### fázisok:

- Inicializálás
- Felismerés
- Összehasonlítás
- Keresés
- Rezonancia, bevésődés.

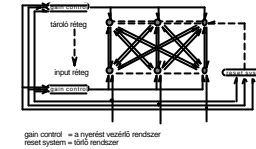
- Inicializálás

- Gain control1 = 1, ha input jelet adunk a bemenetre  
= 0, ha a felismerő, tároló 2. Réteg valamelyik neuronja felismerte az inputot és aktív
- Gain control2 = 1, ha input jelet adunk a bemenetre  
0, ha a vigilance (ismertségi) teszt sikertelen (tiltja a 2. réteg neuronjait és aktivációs szintjüket nullázza)
- $\mathbf{W} \uparrow$  és  $\mathbf{T} \downarrow$  súlymátrix inicializálása:  
 $\mathbf{T} \downarrow$  súlyai= 1, a visszacsatolás teljes értéku  
 $\mathbf{W} \uparrow$  súlyai=  $w_i = 1/(1+N)$   $N$  az inputneuronok száma
- Vigilance (ismertségi) küszöb beállítása:  $0 < \rho < 1$





## ■ Adaptív Rezonancia Elmélet alapú háló ..



### • Felismerési fázis

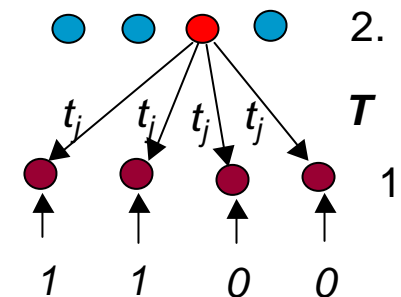
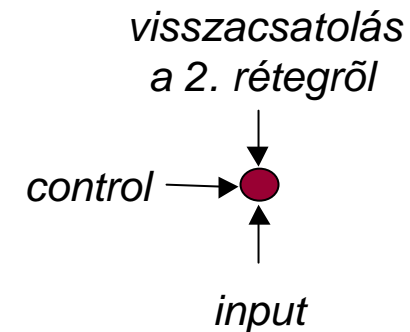
- Input minta ráadva, megpróbálja valamelyik, a 2. rétegbeli neuron által képviselt osztályba besorolni ha a visszaterjesztés után számított ismertségi (vigilance) teszt elég nagy, egyébként új, ismeretlen mintaként kezeli.

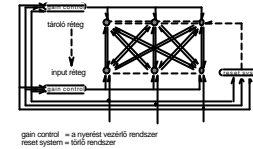
- Az input neuronoknak három bemenete van:  
2/3-os szabály: kimenet=1, ha kettő a háromból aktív, egyébként a kimenet= 0.

- A felismerési fázis a Kohonen féle háló működésére emlékeztet: osztályokba sorolás.

Versengés a 2. rétegben: skalárszorzat az input vektor és a vizsgált 2. rétegbeli neuron súlyai között: az a felismerő (tároló) neuron nyer, amelyiknél a szorzat, hasonlóság a legnagyobb.

- A nyertes  $j^*$  felismerő (tároló) neuron visszaterjeszti az 1 értékű kimenetét a  $T$  súlyain át az 1. input/összehasonlító rétegre.



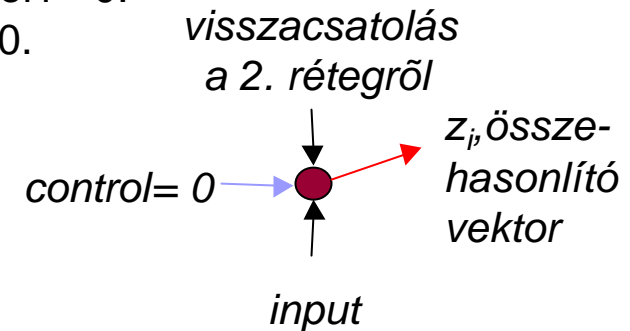


## ■ Adaptív Rezonancia Elmélet alapú háló ..

- **Összehasonlító fázis**

- A 2. rétegbeli nyertes neurontól visszaterjesztett jelek és az input vektor összehasonlítása, eközben  $\text{gain control} = 0$ .  
2/3-os szabály:  $1+1 \Rightarrow 1$ , egyébként  $\Rightarrow 0$ .
- Ismertségi (vigilance) teszt  
Mennyire egyezik  $\mathbf{x}$  és  $\mathbf{z}$  ?

$$\frac{\sum_{i=1}^N t_{ij} x_i}{\sum_{i=1}^N x_i} \stackrel{?}{>} r$$



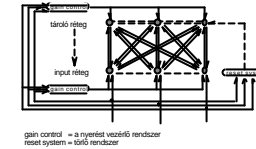
Ha az ismertség elég nagy, akkor az  $\mathbf{x}$  input a nyertes  $j^*$  neuron által képviselt osztályba tartozik, egyébként, ha az ismertség nem elég nagy, belép a keresési fázisba.

A  $\rho$  ismertségi küszöb hatása:

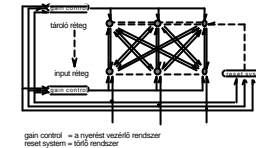
$\rho < 0.4 \rightarrow$  alacsony felbontású osztályozás, mindent egy osztályba sorol

$\rho > 0.8 \rightarrow$  finom felbontás, sok osztály, minden egyed más osztályba kerül.

## ■ Adaptív Rezonancia Elmélet alapú háló ..



- **Keresési fázis**
  - Megpróbál egy nagyobb ismertséget adó felismerő (tároló)neuront találni, de ehhez a **reset system** útján letiltja a korábbi nyertes neuront. Ciklusban ismétli az input felterjesztés, nyerés, visszaterjesztés, összehasonlítás (ismertségi teszt), kis ismertség esetén a nyertes letiltása lépéseket, amíg vagy talál egy jól osztályozó neuront, vagy elfogynak a tároló neuronok. Ha elfogytak, akkor az **x** mintát új mintaként értékeli és egy új tárolóneuront rendel hozzá.
- Rezonancia, bevésoadás
  - Amennyiben az input ismertsége elegendő, a  $j^*$  neuron jól osztályozta az **x** input mintát, a rendszer rezonál, a jelek föl-le áramlanak és a kapcsolatok megerosödnek. Grossberg szerint így módon modellezhető a hosszútávú memória kialakulása.



## Az ART algoritmus

- Inicializálás
  - Súlyok:  $t_{ij}(0)=1$   
 $w_{ij}(0)= 1/(1+N) \quad 1 \leq i \leq N \quad 1 \leq j \leq M$
  - Ismertségi küszöb:  $0 < r < 1$
- Input bemutatása
- Aktivációs szintek számítása a 2. Rétegre

$$\mathbf{m}_j = \sum_{i=1}^N w_{ij}(t) x_i \quad 1 \leq j \leq M$$

- Nyerő kiválasztása;  $j^*$  a nyerő indexe

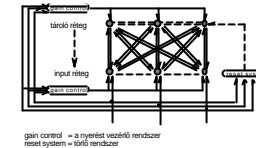
$$\mathbf{m}_{j^*} = \max_j (\mathbf{m}_j)$$

- Ismertségi teszt

$$\frac{\sum_{i=1}^N t_{ij^*}(t) x_i}{\sum_{i=1}^N x_i} > r$$

Igen  $\rightarrow$  goto 7

Nem  $\rightarrow$  goto 6

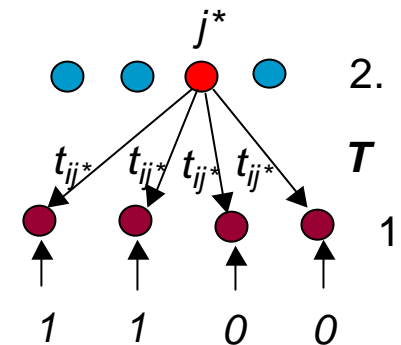


## ■ Az ART algoritmus

- Legjobb egyezés ( $j^*$ ) letiltása a Reset system által.
  - A  $j^*$  neuron kimenetét  $-$ -ra állítani, majd
  - Goto 3.
- A nyertes neuron elfogadása osztályozóként, súlyok ennek megfelelő módosítása (tanulás):

$$t_{ij}(t+1) = t_{ij}(t)x_i \quad 1 \leq i \leq N$$

$$w_{ij}(t+1) = \frac{t_{ij}(t)x_i}{0.5 + \sum_{i=1}^N t_{ij}(t)x_i}$$



- Ismétlés  
Minden letiltott neuron engedélyezése, majd ugrás 2.-re (új inputra).

## ■ Esettanulmány mesterséges neuronháló alkalmazására Folyamatfelügyelet és –szabályozás

A **folyamatfelügyelet** problémái mindennaposak és számtalan alakban jelenhetnek meg. Lényegük egy folyamat bizonyos jellemzőinek **folytonos megfigyelése** azzal a céllal, hogy a folyamat nem megfelelő alakulásakor fellépő jellemző-változásokat azonnal észleljék és a szükséges beavatkozásokat megtehessék. A neurális hálókra alapozott állapotfelügyelet olyan esetekben is alkalmazható, melyeknél a hagyományos állapotfelügyeleti módszerek nem váltak be, illetve a hagyományos eszközökkel kivitelezett megoldásokhoz képest gyorsabb működésű és nagyobb intelligenciával bíró megoldást jelentenek.

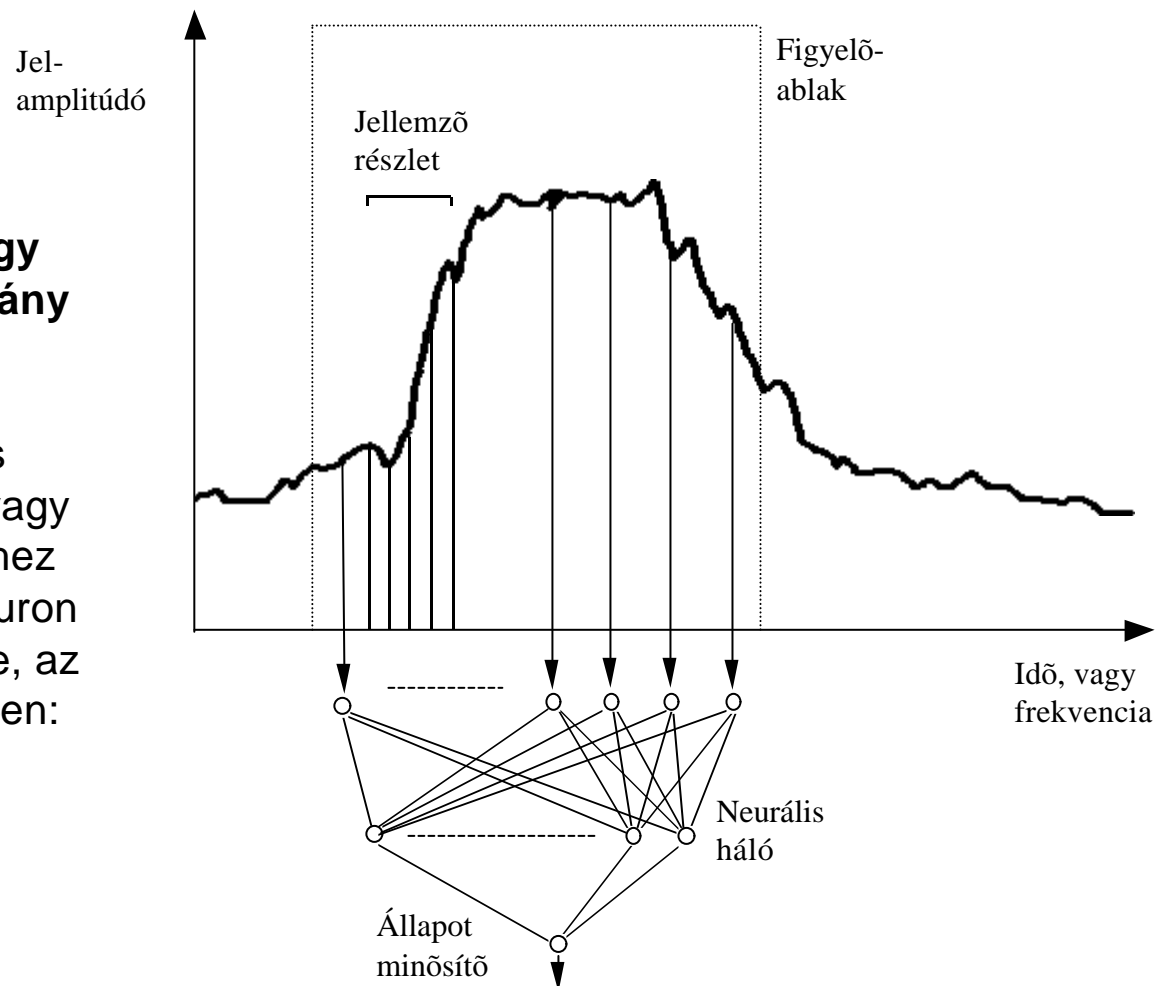
### Ipari szerelőrobot megfogási és szerelési műveleteinek felügyeletére alkalmazott neurális alapú folyamatfelügyelő rendszer

- Mivel rendszerint gyors válaszreakcióra van szükség, a folyamatfelügyeletben és a szabályozásban általánosan elterjedt a többrétegű előrecsatolt neurális hálók alkalmazása. Ezek többféle szerepkörben is megjelenhetnek:
  - néha speciális szűrőként, vagy jellemző-felismerőként, vagy
  - folyamat-modell hordozójaként, illetve
  - neurális elvű szabályzóként jelenik meg

## ■ Folyamatfelügyelet és –szabályozás

A felügyelt folyamat szenzorainak normalizált jeleivel, mint input jelekkel működő neurális hálózat egyetlen **output** neuronja **1** értékével jelezheti a **normális** folyamatállapotot, **0** értékkel pedig a **hiba** fellépését.

A neurális háló alkalmazható **stacionér időtartomány, vagy frekvenciatartomány** mintázatának beazonosítására, amikor is az egyes időpillanatokhoz, vagy frekvenciaértékekhez más-más input neuron van hozzárendelve, az ábrának megfelelően:



## ■ Folyamatfelügyelet és –szabályozás

A neurális háló bemenetét a **figyeloablak** egy konstans pozíciójától kezdve választjuk. Azok a görberészletek, melyek fontosabb információt tartalmaznak, finomabb közökkel **mintavételezve** adhatók a neuronokra. A **neuronkimenet** általában két kategóriára osztja a megfigyelt állapotokat: **normális**, vagy **deviáns**. Pl. egy speciális függvénymintához betanított deviáns kimenet jelezhet egy szerszámtörést egy megfigyelt forgácsolási folyamatnál.

A **betanító görbeseregnek** tartalmaznia kell a normális üzemállapot esetén mérhető alakokat csakúgy, mint a hibák esetén fellépőket.

Speciális **jelszurést** valósíthatunk meg, ha **ido ben mozgó megfigyelo-ablakot** alkalmazunk. Mivel az ablak mindig az utolsó N inputértéket tartalmazza, és ezeket minden megfigyelo **ciklusban léptetjük**, a függvényminta a neurális háló számára mozgónak tunik. Ilyen **idofüggo minta** esetében nemcsak a mintapárok betanulása, hanem néhány mintából álló mintahalmaz megtanulása válik szükségessé, melyek idoben közrefogják a központi jellegzetes mintát.

Mozgó ablak esetében **keskenyebb figyeloablak** is elegendo lehet, ha a minta viszonylag egyszeru.



## ■ Folyamatfelügyelet és –szabályozás ..

Az általunk vizsgált példa egy **szerelőrobot folyamatos felügyeletét** mutatja be. **Akusztikus** (hang-) **érzékelő** alkalmaz az alkatrész és a robot markolója között létrejövő hangok észlelésére és ezen keresztül a megfogás sikerességének jelzésére, végig a szerelés időtartama alatt.

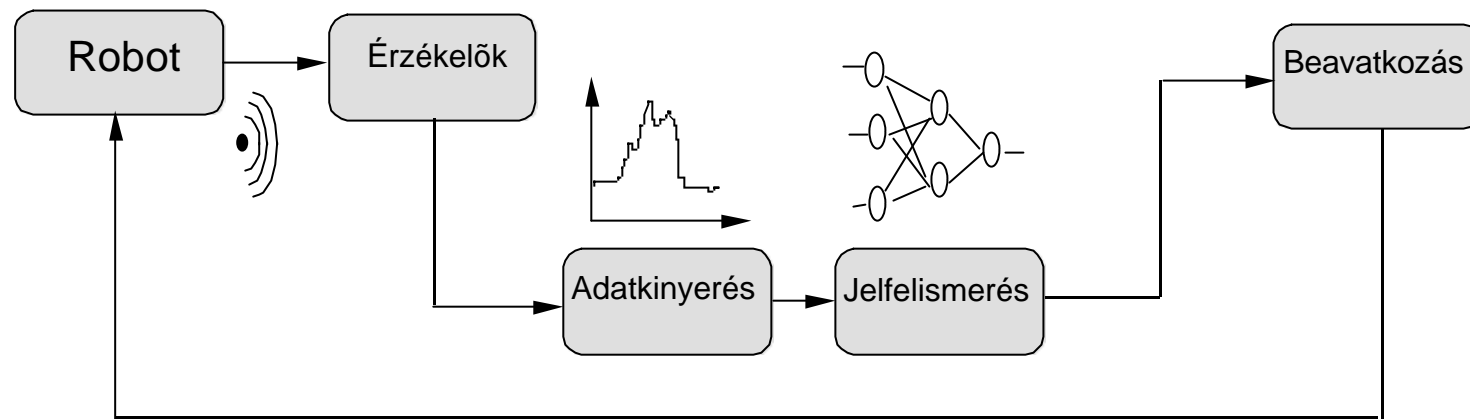
A felügyeleti rendszer egy **négylépéses modellt** alkalmaz az időtartomány-függő akusztikus jelminták **valós idejű kiértékelésére**.

A különféle **deviáns eseteket** kísérő *hangjelenségek* felismerése lehetővé teszi korrekciós rutinok meghívását, a veszélyes helyzetek elkerülését. Minden maradó alakváltozás a munkadarabon, de még a mikroszkopikus felületi deformációk is észlelhető hangokkal járnak együtt. A különféle **robotoperációkat kísérő zajok** jellemzik a tevékenységeket és alkalmasak folyamatos felügyelet input jeleiként.



## ■ Folyamatfelügyelet és –szabályozás ..

A robotmuveletek időtartománybeli folyamatos felügyeletének sémája :



A robot markolója és a megfogott munkadarab által keltett hangot az érzékelő felfogja, átalakítás után a betanított neuronhálóra kerül, amely intelligensen megvizsgálja. Amennyiben a betanított deviáns mintákhoz közelálló az inputjel, a kimenet 0 szintje jelzi a beavatkozás szükségességét.

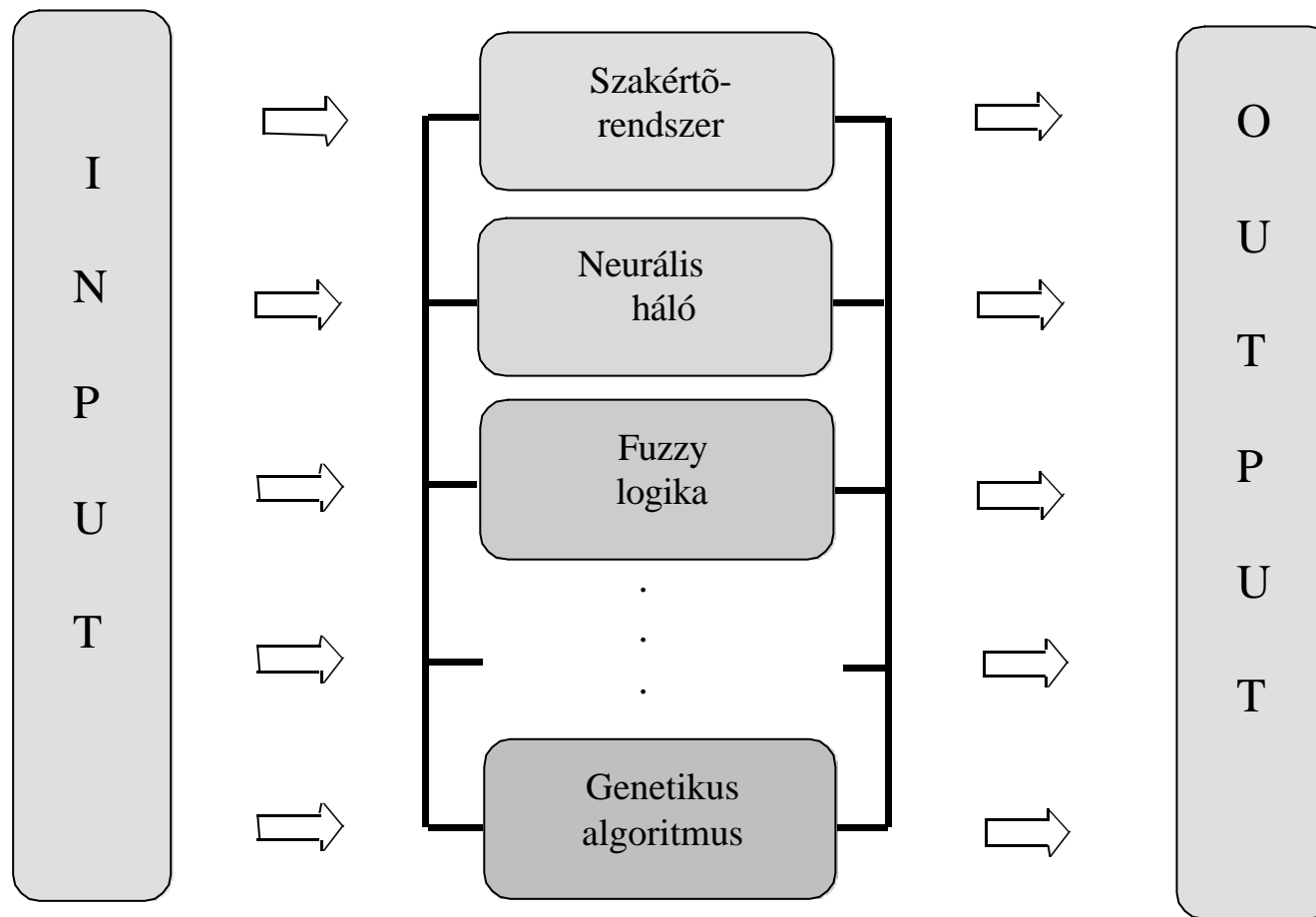
## ■ Hibrid intelligens rendszerek

- Az intelligens rendszerek következő generációja **integrálni** fogja a *szakértőrendszereket*, a *mesterséges neurális rendszereket*, a *fuzzy-logikát*, a *bizonytalanság-kezelést* és a *genetikus algoritmusokat*, több más technika mellett. Ez a megközelítés válasz a tisztán szimbolikus, numerikus, vagy elosztott rendszerek gyenge hatásfokára és azokra az eljárásokra, melyek komplex és nemprecíz adatok és a valós világ problémáinak megoldására vonatkozó célok kezelésére születtek
- A **két tervezési módszertan**, mely az ismeretbázisú rendszerekből származik és lehetővé teszi többféle mesterséges intelligencia módszer egyesítését a
  - **hierarchikus megközelítés** és a
  - **falitábla** módszer.

## ■ A hierarchikus megközelítés

- A **hierarchikus megközelítés** intelligens rendszerek tervezésére különböző **rétegek**, illetve **szintek** használatát jelenti. A magasszintu célokat felbontják több elemi, vagy egyszerűbb rész célra, vagy alproblémára. A felbontást mindaddig folytatják, míg nem könnyen megvalósítható rész célokhoz nem érnek.
- A különböző szintu modulok információi rendelkezésre állnak a magasabb szinteken rendszerint egy olyan **kommunikációs mechanizmus** révén, amely lehetővé teszi a visszacsatolás kialakulását.
- A kontrol általában **felülrol-lefelé** irányuló. Ez lehetővé teszi a viszonylag egyenesvonalú rendszertervezést. Ha a rendszer sokszintu, akkor nehezebb a visszacsatolás kialakítása. Ez kompromisszumokra kényszeríti a tervezőket.

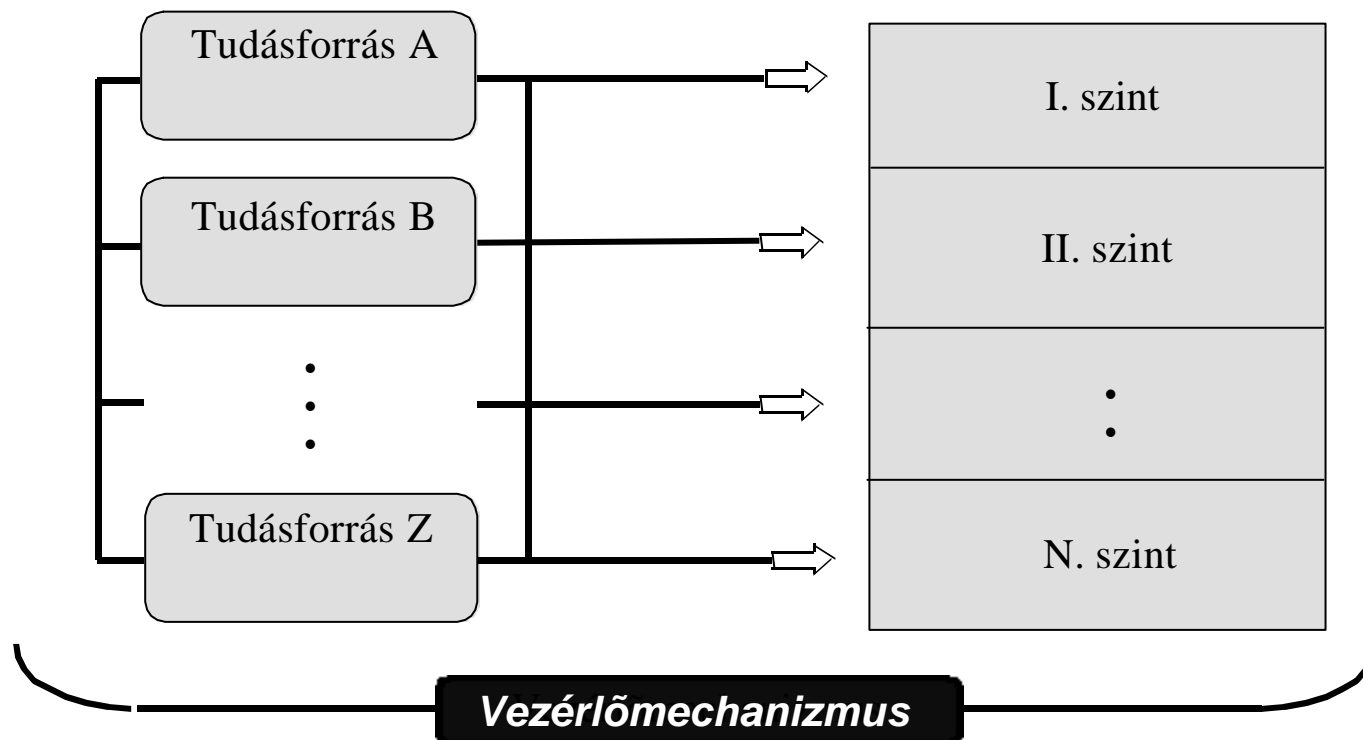
## ■ Hibrid intelligens rendszerek ..



Hibrid intelligens rendszer

## ■ A falitábla rendszer

- A **falitábla rendszer** egy erősen struktúrált, opportunista problémamegoldó rendszer, amely **a tudást tartományokba és megoldásterekbe szervezi**. Alapvetően három részéről szokás beszélni:
  - a tudásforrások,
  - a falitábla adatszerkezete és
  - a vezérlőmechanizmus



## ■ A falitábla rendszer ..

- **A tudásforrások** tartalmazzák a probléma megoldásához szükséges **tudást**. Ez szeparálva van független részekre, melyek a **deklaratív** és a **procedurális** tudást hordozzák. Ezek a modulok a megoldástér (a falitábla) ismeretanyagát determinisztikus, vagy heurisztikus algoritmusokkal megváltoztatják. Ezek a változtatások akkor zajlanak, ha egy kis részmegoldást képesek adni a falitábla tartalmához. Mindehhez tudniuk kell, mikor módosíthatják a falitáblát.
- **A falitábla adatstruktúrája** tárolja a rendszerállapot információkat egy globális adatbázisban. Mint ilyen, tartalmazza az összes teljes, vagy részleges lehetséges megoldást. Ez a megoldástér egy, vagy több alkalmazásfüggő elemzészintre tagolódik, rendszerint hierarchikusan, ahol az információ minden egyes szinten részmegoldást jelent. Minden **kommunikáció** a tudásforrások között **a falitábla-adatbázison keresztül** zajlik, és a tudáselemek szintén használják az erőforrásukat annak figyelésére, hogy mikor vehetnek részt a megoldástér alakításában. Ilyen típusú rendszerben a célállapot egy részproblémára vonatkozó megoldás.
- **A vezérlő mechanizmus** szervezi a tudásforrások válaszait a falitábla felé, valamint meghatározza a rendszer figyelmének irányát. Ez a mechanizmus hivatott leállítani a problémamegoldást is.