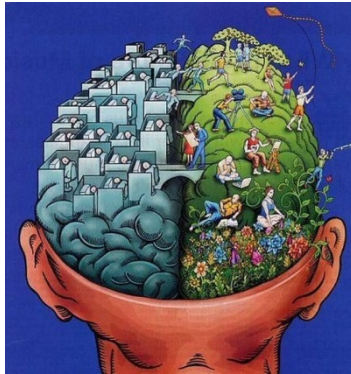




Budapesti Műszaki és Gazdaságtudományi Egyetem Méréstechnika és Információs rendszerek Tanszék



Megerősítőes tanulás

Előadó:

Hullám Gábor

Előadás anyaga:

Dobrowiecki Tadeusz

Szekvenciális döntési problémából indulunk

Markov döntési folyamat (MDF)

S_0

$T(s, a, s')$

$R(s)$, v. $R(s, a, s')$

$$U(s) = R(s) + \gamma \max_a \sum_{s'} T(s, a, s') U(s')$$

$$\pi(s) \leftarrow U(s)$$

$$U_{i+1}(s) \leftarrow R(s) + \gamma \max_a \sum_{s'} T(s, a, s') U_i(s') \quad \pi(s) \leftarrow U_i(s)$$

$$\pi^*(s) = \arg \max_a \sum_{s'} T(s, a, s') U(s') \quad \text{Optimális eljárás mód}$$

Megerősítéses tanulás

Kezdőállapot:

ismert

Állapotátmenet-modell:

nem ismert, legfeljebb a tapasztalat

Jutalomfüggvény:

nem ismert, legfeljebb ahogy jön

Optimális eljárás mód

megtanulni, mindennek ellenére?!



Megerősítéses tanulás

Pl. sakkjáték: tanító nélkül is van visszacsatolás a játék végén:
nyert vagy **vesztett** = +/- **jutalom**, azaz **megerősítés**.

... a játék végén = a cselekvés szekvencia végén

Megerősítés: milyen gyakran? milyen erős? milyen értékű?

A feladat:

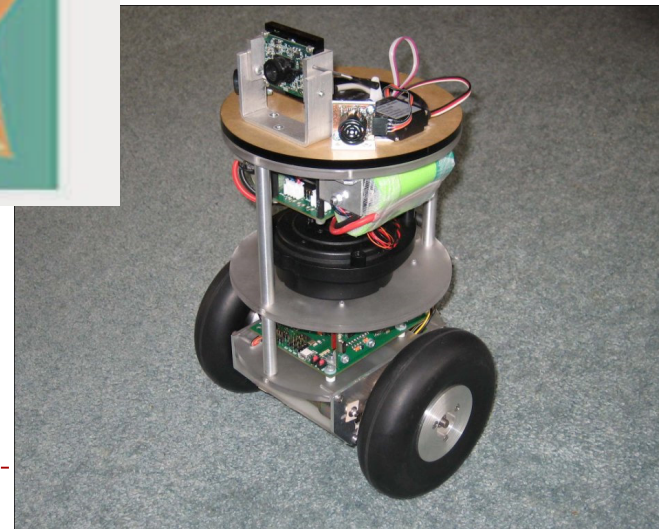
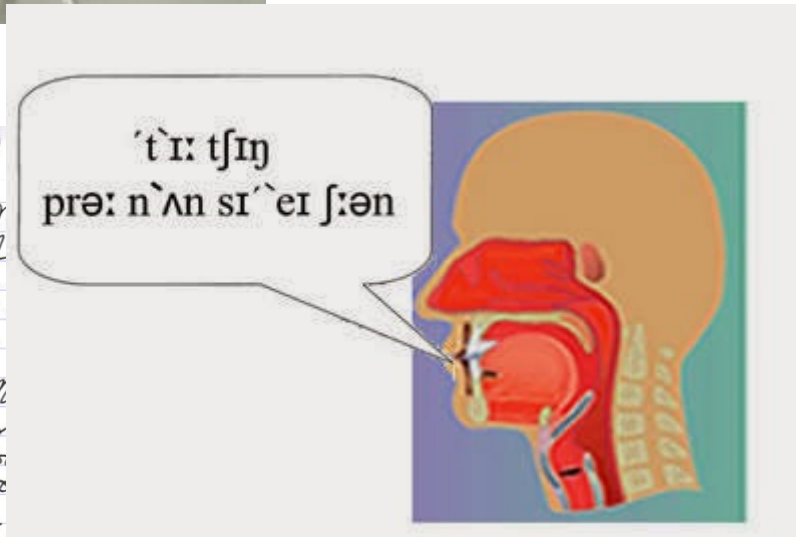
(ritka) jutalmakból megtanulni egy sikeres ágens-függvényt
(optimális eljárásmód: melyik állapot, melyik cselekvés hasznos).

Nehéz: az információhiány miatt az ágens sohasem tudja, hogy
mik a jó lépések, azt sem, **melyik jutalom melyik cselekvésből**
ered.





m m m m m m m m
 m m m m m m m m
 n n n n n n n n
 n n n n n n n n
 Moon Moon Moon Moon
 Moon Moon Moon Moon
 Moon Moon Moon Moon
 Moon Moon Moon Moon
 Noon Noon Noon Noon
 Noon Noon Noon Noon
 Noon Noon Noon Noon
 Noon Noon Noon Noon
 a a a a a a a a
 d d d d d d d d
 q q q q q q q q
 q q q q q q q q
 q q q q q q q q
 q q q q q q q q
 gadding gadding gadding
 gadding gadding gadding
 gadding gadding gadding
 gadding gadding gadding



Ágens tudása:

induláskor **tudja már** a környezetet és a cselekvéseinek hatását,
vagy pedig még ezt is **meg kell tanulnia**.

Megerősítés:

csak a **végállapotban**, vagy menet közben **bármelyikben**.

Ágens:

passzív tanuló: figyeli a világ alakulását és tanul.

aktív tanuló: a megtanult információ birtokában cselekednie is kell.
felfedező ...

Ágens kialakítása: megerősítés $\rightarrow$ hasznosság leképezés

$U(s)$ **hasznosság** függvény tanulása, ennek alapján a cselekvések
eldöntése, hogy az elérhető hasznosság várható értéke max legyen
(környezet/ágens modellje **szükséges**)

$Q(a, s)$ **cselekvés érték** függvény (állapot-cselekvés párok) tanulása,
valamilyen várható hasznót tulajdonítva egy adott helyzetben egy adott
cselekvésnek (környezet/ágens modell **nem szükséges**, közben tanult)
– **Q tanulás (model-free)**



internal state



reward

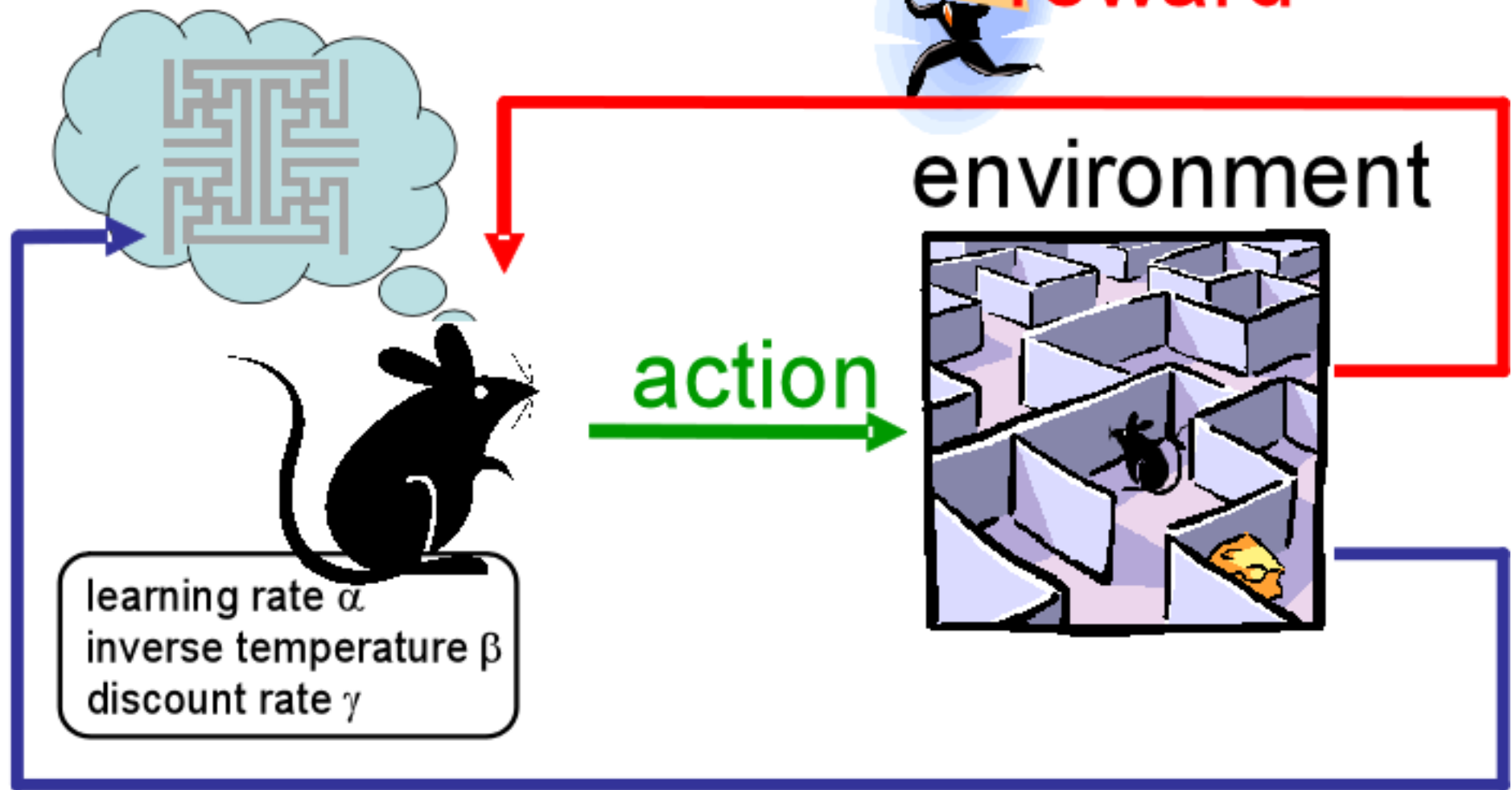
environment

action



learning rate α
inverse temperature β
discount rate γ

observation



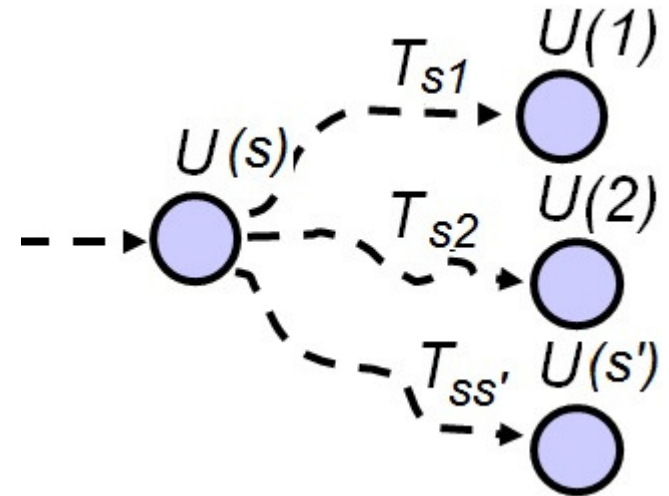
Fontos egyszerűsítés (ld. a MDF): a sorozat hasznossága
= a sorozat állapotaihoz rendelt hasznosságok összege.
= a hasznosság **additív**

Az állapot **hátralevő-jutalma (reward-to-go)**:
azon jutalmak összege, amelyet akkor kapunk, ha az adott állapotból
valamelyik végállapotig eljutunk.

Egy állapot várható hasznossága = a hátralevő-jutalom várható értéke

$$U^\pi(s) = E[\sum_t \gamma^t R(s_t) | \pi, s_0 = s]$$

$$U^\pi(s) = R(s) + \gamma \sum_{s'} T(s, \pi(s), s') U^\pi(s')$$



Adaptív Dinamikus Programozás

Az állapotátmeneti valószínűségek T_{ij} a megfigyelt gyakoriságokkal becsülhetők.

Amint az ágens megfigyelte az összes állapothoz tartozó jutalom értéket, a hasznosság-értékek a következő **egyenletrendszer megoldásával** kaphatók meg:

$$U^\pi(s) = R(s) + \gamma \sum_{s'} T(s, \pi(s), s') U^\pi(s')$$

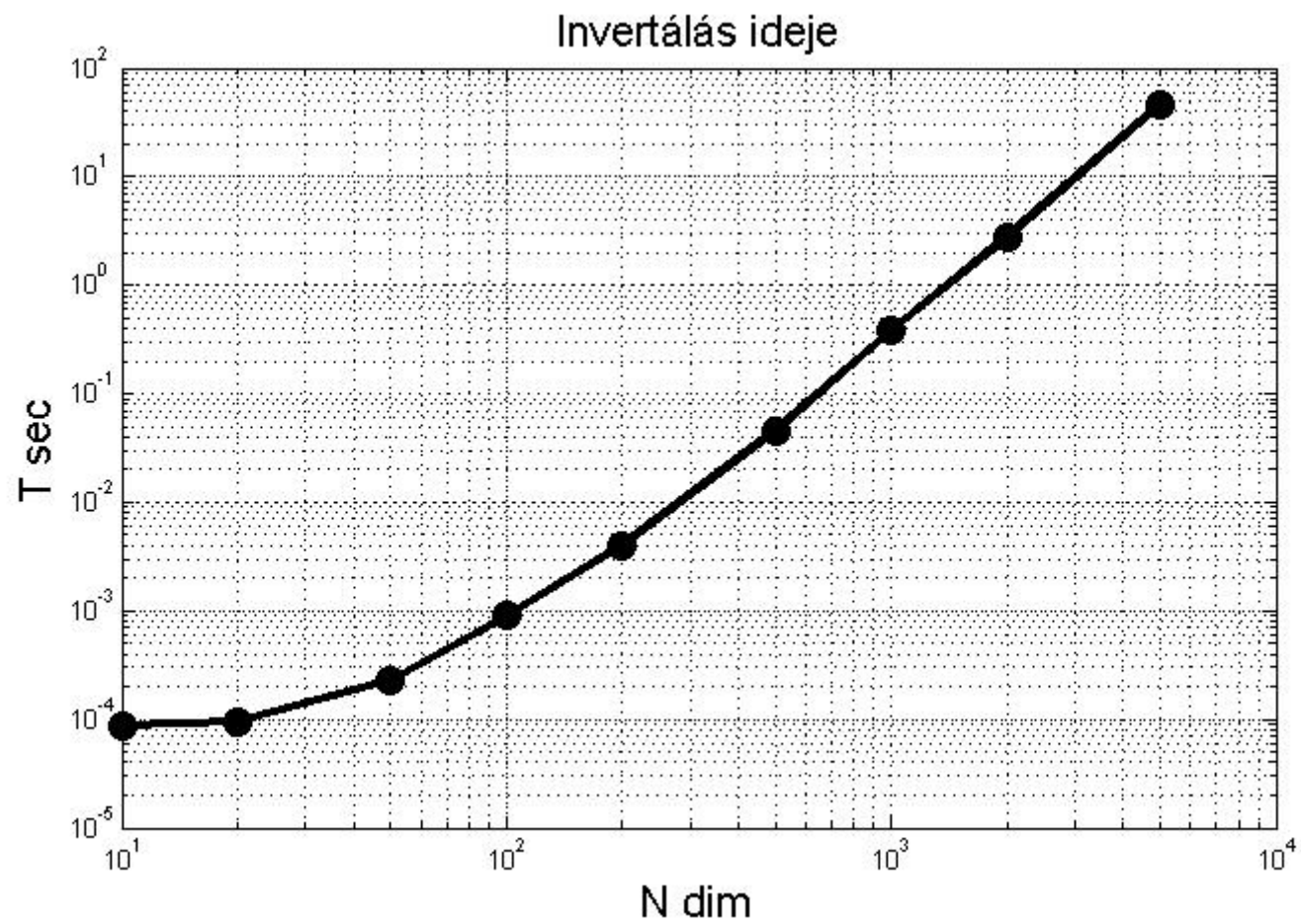
megfigyelt $\nearrow$ $\nwarrow$ megtanult

$$\begin{pmatrix} U(1) \\ U(2) \\ U(3) \\ \dots \\ U(N) \end{pmatrix} = \begin{pmatrix} R(1) \\ R(2) \\ R(3) \\ \dots \\ R(N) \end{pmatrix} + \gamma \begin{pmatrix} T_{11} & T_{12} & T_{13} & \dots & T_{1N} \\ T_{21} & T_{22} & T_{23} & \dots & T_{2N} \\ T_{31} & T_{32} & T_{33} & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ T_{N1} & T_{N2} & \dots & \dots & T_{NN} \end{pmatrix} \begin{pmatrix} U(1) \\ U(2) \\ U(3) \\ \dots \\ U(N) \end{pmatrix}$$

$$\mathbf{U} = \mathbf{R} + \gamma \mathbf{T} \mathbf{U} \quad \mathbf{U} = (\mathbf{I} - \gamma \mathbf{T})^{-1} \mathbf{R}$$



és ha nem megy?



Az időbeli különbség (IK) tanulás (TD – Time Difference)

$$U(s) \leftarrow U(s) + \alpha(R(s) - U(s))$$

Monte Carlo mintavétel: használjuk fel a megfigyelt állapotátmeneteket a megfigyelt visszaterjesztett megerősítésekkel
- ki kell várni az epizód végét.

Alapötlet: használjuk fel a megfigyelt állapotátmenetekenél a jutalom becsült értékét:

TD(0)-különbség (a jutalom becslése)

TD(0) -hiba:

$$R(s) + \gamma U(s')$$

$$\delta(s) = R(s) + \gamma U(s') - U(s)$$

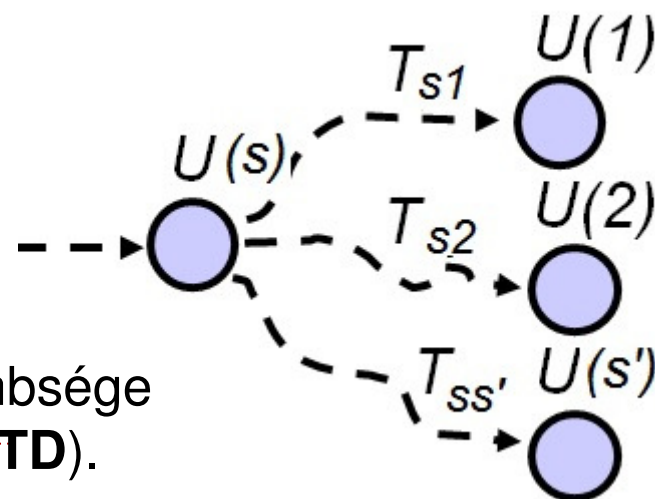
$$U(s) \leftarrow (1 - \alpha) U(s) + \alpha \delta(s) =$$

$$U(s) + \alpha(R(s) + \gamma U(s') - U(s))$$

α - **bátorsági faktor**, tanulási tényező

γ - **leszámoltatási** tényező (ld. MDF)

Az egymást követő állapotok hasznosság-különbsége
időbeli különbség (IK) (temporal difference, TD).



Az összes időbeli különbség eljárásmód alapötlete

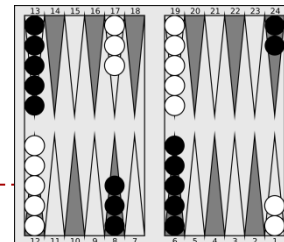
1. korrekt hasznosság-értékek esetén **lokálisan fennálló feltételek rendszere:**

$$U^\pi(s) = R(s) + \gamma \sum_{s'} T(s, \pi(s), s') U^\pi(s')$$

2. **módosító egyenlet**, amely a becsléseinket ezen „**egyensúlyi**” **egyenlet irányába** módosítja:

$$U(s) \leftarrow U(s) + \alpha(R(s) + \gamma U(s') - U(s))$$

3. Csak a **(A)DP** módszer használta fel a **modell teljes** ismeretét.
Az **IK** az állapotok közt fennálló kapcsolatokra vonatkozó információt használja, de csak azt, amely az **aktuális tanulási sorozatból származik**.
Az **IK** változatlanul **működik előzetesen ismeretlen környezet esetén is**.



Ismeretlen környezetben végzett aktív tanulás

Döntés: melyik cselekvés? a cselekvésnek mik a kimeneteleik?

hogyan hatnak az elért jutalomra?

A környezeti modell: a többi állapotba való **átmenet valószínűsége egy adott cselekvés esetén.**

$$U(s) = R(s) + \gamma \max_a \sum_{s'} T(s, a, s') U(s')$$

Az állapottér felderítése - milyen cselekvést válasszunk? **Nehéz probléma**

Helyes-e azt a cselekvést választani, amelynek a jelenlegi hasznosság-becslés alapján legnagyobb a várható hasznossága?

Ez figyelmen kívül hagyja a **cselekvésnek a tanulásra gyakorolt hatását.**

Döntésnek kétféle hatása van:

1. Jutalmat eredményez a **jelenlegi** szekvenciában.
2. Befolyásolja az észleléseket, és ezáltal az ágens **tanulási képességét**
– így jutalmat eredményezhet a **jövőbeni** szekvenciákban.



Az állapottér felderítése

Kompromisszum: a **jelenlegi jutalom**, amit a pillanatnyi hasznosság-becslés tükröz, és a **hosszú távú előnyök** közt.

Két megközelítés a cselekvés kiválasztásában:

„**Hóbortos, Felfedező**„: véletlen módon cselekszik, annak reményében, hogy végül is felfedezi az egész környezetet

„**Mohó**„: a jelenlegi becslésre alapozva maximalizálja a hasznót.

Hóbortos: képes jó hasznosság-becsléseket megtanulni az összes állapotról.
Sohasem sikerül fejlődnie az optimális jutalom elérésében.

Mohó: gyakran talál egy jó utat. Utána ragaszkodik hozzá, és soha nem tanulja meg a többi állapot hasznosságát.

Ágens addig legyen hóbortos, amíg kevés fogalma van a környezetről, és legyen mohó, amikor a valósághoz közeli modellel rendelkezik.

Létezik-e optimális felfedezési stratégia?

Ágens súlyt kell adjon azoknak a cselekvéseknek, amelyeket még nem nagyon gyakran próbált, a kis hasznosságúnak gondolt cselekvéseket elkerülje.



felfedezési függvény

$$f(u, n) = \begin{cases} R^+ & \text{ha } n < N_e \\ u & \text{különben} \end{cases}$$

mohóság $\leftrightarrow$ kíváncsiság

$f(u, n)$ u -ban monoton növekvő,
 n -ben monoton csökkenő

$$U^+(s) \leftarrow R(s) + \gamma \max_a f(\sum_{s'} T(s, a, s') U^+(s'), N(a, s))$$

$U^+(i)$: az i állapothoz rendelt hasznosság optimista becslése

$N(a, i)$: az i állapotban hányszor próbálkoztunk az a cselekvéssel

R^+ a tetszőleges állapotban elérhető legnagyobb jutalom optimista becslése

Az a cselekvés, amely felderítetlen területek *felé* vezet, nagyobb súlyt kap.

ϵ -mohóság: az ágens ϵ valószínűséggel véletlen cselekvést választ, ill. $1-\epsilon$ valószínűséggel mohó

Boltzmann-felfedezési modell

egy a cselekvés megválasztásának valószínűsége

egy s állapotban:

$$P(a, s) = \frac{e^{\text{hasznosság}(a, s)/T}}{\sum_{a'} e^{\text{hasznosság}(a', s)/T}}$$

a T „hőmérséklet” a két véglet között szabályoz.

Ha $T \rightarrow \infty$, akkor a választás tisztán (egyenletesen) véletlen,

ha $T \rightarrow 0$, akkor a választás mohó.

A cselekvés-érték függvény tanulása

A cselekvés-érték függvény egy adott állapotban választott adott cselekvéshez egy várható hasznosságot rendel: **Q-érték**

A Q-értékek fontossága:

$$U(s) = \max_a Q(a, s)$$

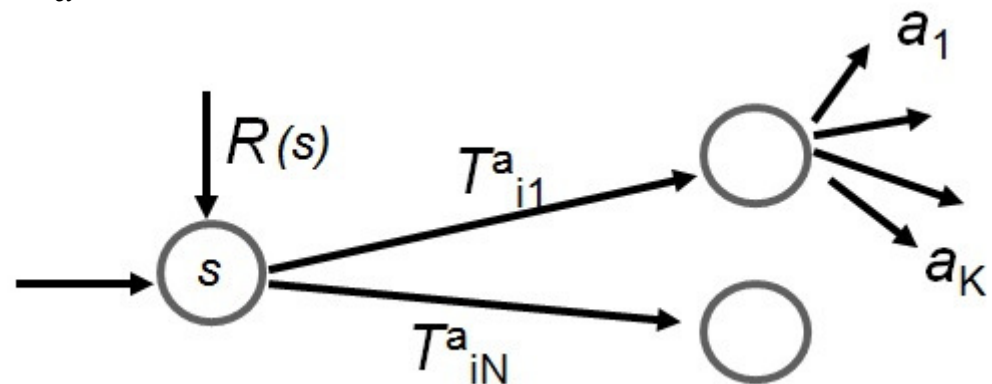
a feltétel-cselekvés szabályokhoz hasonlóan lehetővé teszik a döntést

modell használata nélkül,

ellentétben a feltétel-cselekvés szabályokkal, **közvetlenül** a jutalom visszacsatolásával **tanulhatók**.

Mint a hasznosság-értékekénél, felírhatunk egy kényszer egyenletet, amely **egyensúlyi állapotban**, amikor a **Q-értékek korrektek**, fenn kell álljon:

$$Q(a, s) = R(s) + \gamma \sum_{s'} T(s, a, s') \max_{a'} Q(a', s')$$



A cselekvés-érték függvény tanulása

Ezt az egyenletet közvetlenül felhasználhatjuk egy olyan iterációs folyamat módosítási egyenleteként, amely egy adott modell esetén a pontos Q-értékek számítását végzi.

Az **időbeli különbség** viszont nem igényli a modell ismeretét. Az **IK** módszer **Q-tanulásának** módosítási egyenlete:

$$Q(a, s) \leftarrow Q(a, s) + \alpha (R(s) + \gamma \max_{a'} Q(a', s') - Q(a, s))$$

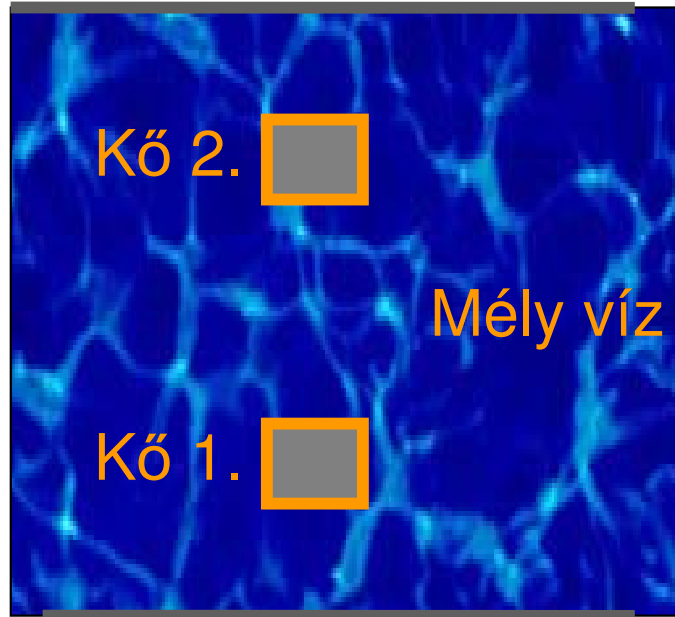
SARSA Q-tanulás (State-Action-Reward-State-Action)

$$Q(a, s) \leftarrow Q(a, s) + \alpha [R(s) + \gamma Q(a', s') - Q(a, s)]$$

(a' megválasztása pl. Boltzmann felfedezési modellből)



Folyópart B



Folyópart A

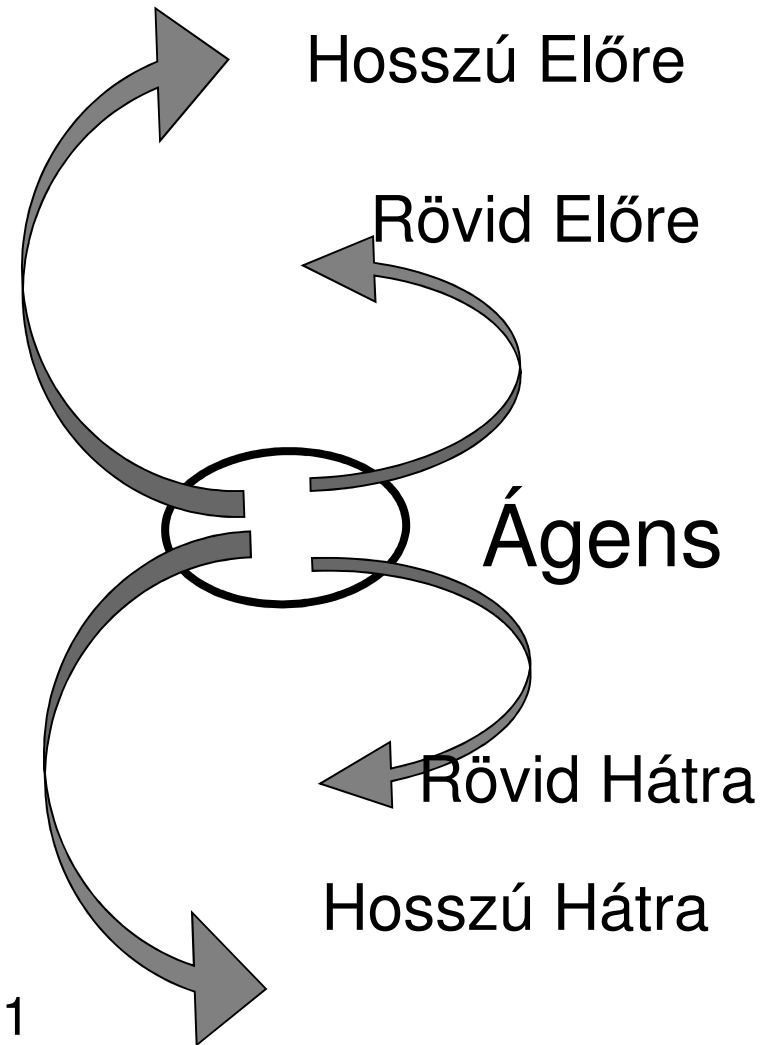
Tanuló szekvencia:

RE → HE → RE → +1 (száraz lábbal át)

RE → HE → HH → HE → HH → RH → HE → -1

RE → HE → RH → -1 (megfürdött)

.....



Part B

Kő 2.



$$Q(a, s) \leftarrow Q(a, s) + \alpha (R(s) + \gamma \max_{a'} Q(a', s') - Q(a, s))$$

Kő 1.



Part A

Cselekvés

		HH	RH	RE	HE
Állapot	Part A	0	0	-0.035	-0.877
	Kő 1.	-0.520	-0.550	-0.835	0.555
	Kő 2.	-0.204	-0.897	1.180	1.158



Part B

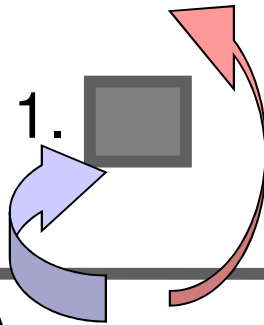
Kő 2.



Kő 1.

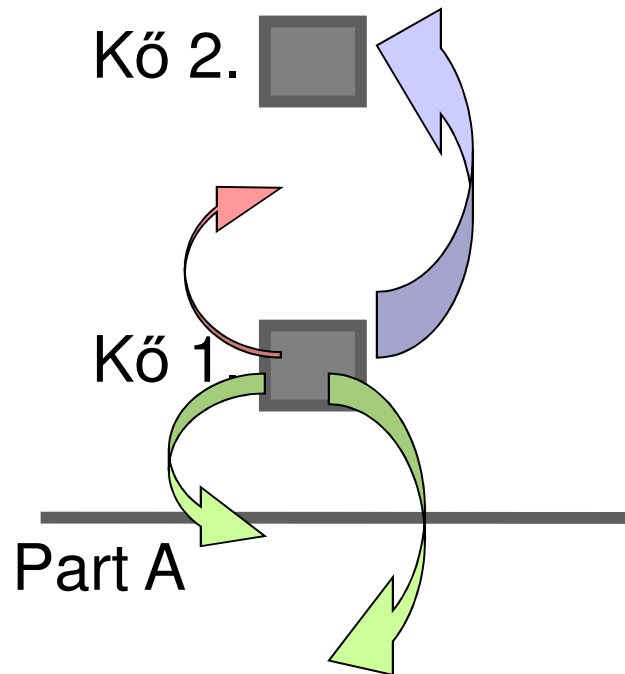


Part A

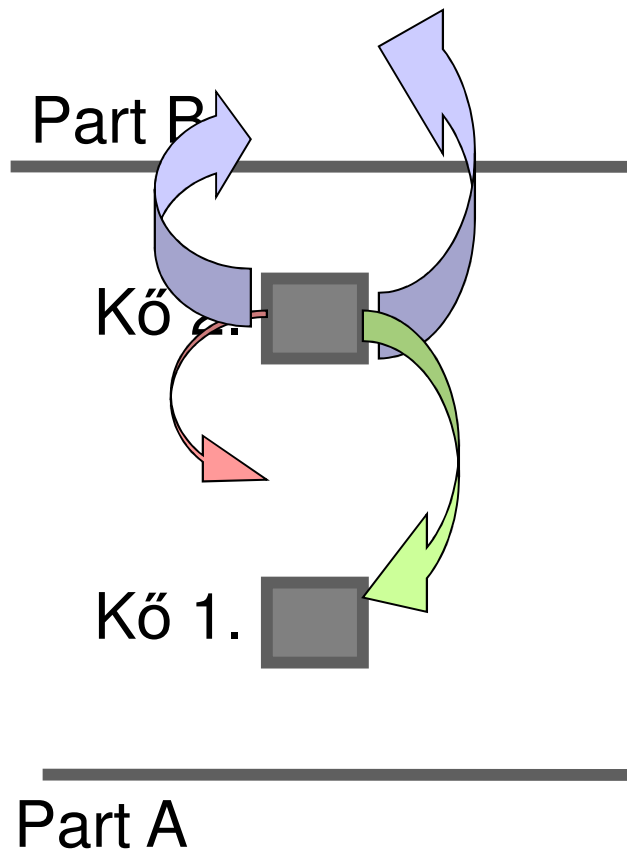


	HH	RH	RE	HE
Part A	0	0	-0.035	-0.877
Kő 1.	-0.520	-0.550	-0.835	0.555
Kő 2.	-0.204	-0.897	1.180	1.158

Part B



	HH	RH	RE	HE
Part A	0	0	-0.035	-0.877
Kő 1.	-0.520	-0.550	-0.835	0.555
Kő 2.	-0.204	-0.897	1.180	1.158



	HH	RH	RE	HE
Part A	0	0	-0.035	-0.877
Kõ 1.	-0.520	-0.550	-0.835	0.555
Kõ 2.	-0.204	-0.897	1.180	1.158

A megerősítéssel tanulás általánosító képessége

Eddig: ágens által tanult hasznosság: **táblázatos** formában reprezentált
= **explicit reprezentáció**

Kis állapotterek: elfogadható, de a konvergencia idő és (ADP esetén) az iterációnkénti idő gyorsan nő a tér méretével.

Gondosan vezérelt közelítő ADP: 10000, vagy ennél is több.

De a való világhoz közelebb álló környezetek szóba se jöhetnek.

(a sakk stb. - állapottere 10^{50} - 10^{120} nagyságrendű)

(az összes állapotot látni, hogy megtanuljunk játszani?!)

Egyetlen lehetőség: a függvény **implicit reprezentációja**:

Pl. egy játékban a táblaállás tulajdonságainak valamilyen halmaza $f_1, \dots, f_n$.

Az állás becsült hasznosság-függvénye:

$$\hat{U}_\theta(s) = \theta_1 f_1(s) + \theta_2 f_2(s) + \dots + \theta_n f_n(s)$$

C. Shannon, **Programming a Computer for Playing Chess**, Philosophical Magazine, 7th series, 41, no. 314 (March 1950): 256-75

A.L. Samuel, **Some Studies in Machine Learning Using the Game of Checkers.**

B.II-Recent Progress, IBM. J. 3, 211-229 (1959), kb. 20 tulajdonság

Deep Blue (Feng-Hsiung Hsu) kb. 8000 tulajdonság, spec. sakk-csíp

A hasznosság-függvény n értékkel jellemezhető, pl. 10^{120} érték helyett. Egy átlagos sakk kiértékelő függvény kb. 10 súllyal épül fel, tehát *hatalmas* tömörítést értünk el.

Az implicit reprezentáció által elért tömörítés teszi lehetővé, hogy a tanuló ágens általánosítani tudjon a már látott állapotokról az eddigiekben nem látottakra.

Az implicit reprezentáció legfontosabb aspektusa nem az, hogy kevesebb helyet foglal, hanem az, hogy lehetővé teszi a **bemeneti állapotok induktív általánosítását**. Az olyan módszerekről, amelyek ilyen reprezentációt tanulnak, azt mondjuk, hogy **bemeneti általánosítást** végeznek.



4 x 3 világ

$$E_j(s) = \frac{1}{2} \left(\hat{U}_\theta(s) - u_j(s) \right)^2$$

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y$$

Jósolt és kísérleti tényleges

$$\theta_i \leftarrow \theta_i - \alpha \frac{\partial E_j(s)}{\partial \theta_i} = \theta_i + \alpha \left(u_j(s) - \hat{U}_\theta(s) \right) \frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}$$

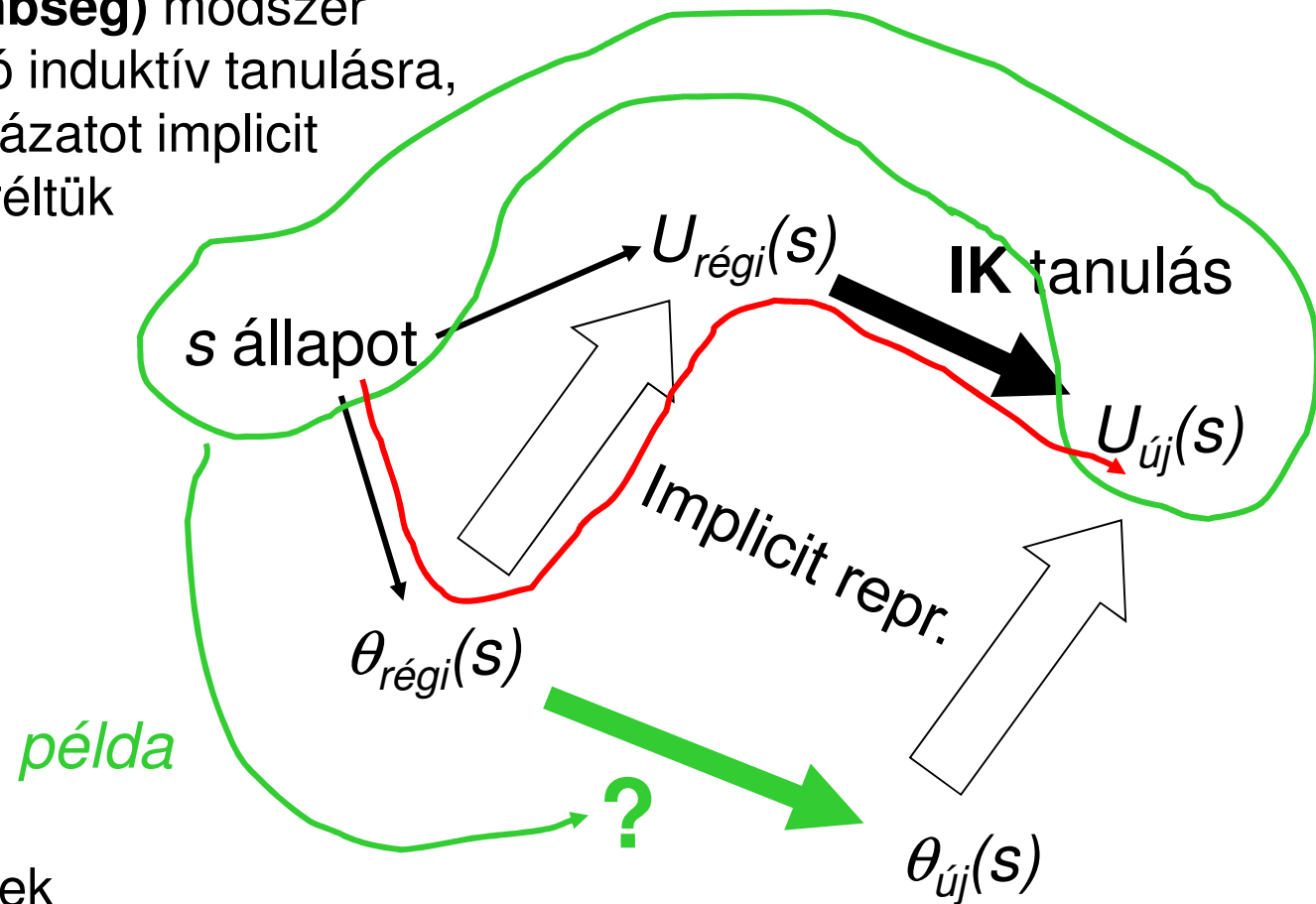
$$\theta_0 \leftarrow \theta_0 + \alpha \left(u_j(s) - \hat{U}_\theta(s) \right)$$

$$\theta_1 \leftarrow \theta_1 + \alpha \left(u_j(s) - \hat{U}_\theta(s) \right) x$$

$$\theta_2 \leftarrow \theta_2 + \alpha \left(u_j(s) - \hat{U}_\theta(s) \right) y$$



Az **IK (időbeli különbség)** módszer szintén alkalmazható induktív tanulásra, ha az U , vagy Q táblázatot implicit reprezentációra cseréltük (pl. neurális háló).



U/Q tanuló gradiensek

$$\theta_i \leftarrow \theta_i + \alpha \left[R(s) + \gamma \hat{U}_\theta(s') - \hat{U}_\theta(s) \right] \frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}$$

$$\theta_i \leftarrow \theta_i + \alpha \left[R(s) + \gamma \max_{a'} \hat{Q}_\theta(a', s') - \hat{Q}_\theta(a, s) \right] \frac{\partial \hat{Q}_\theta(a, s)}{\partial \theta_i}$$