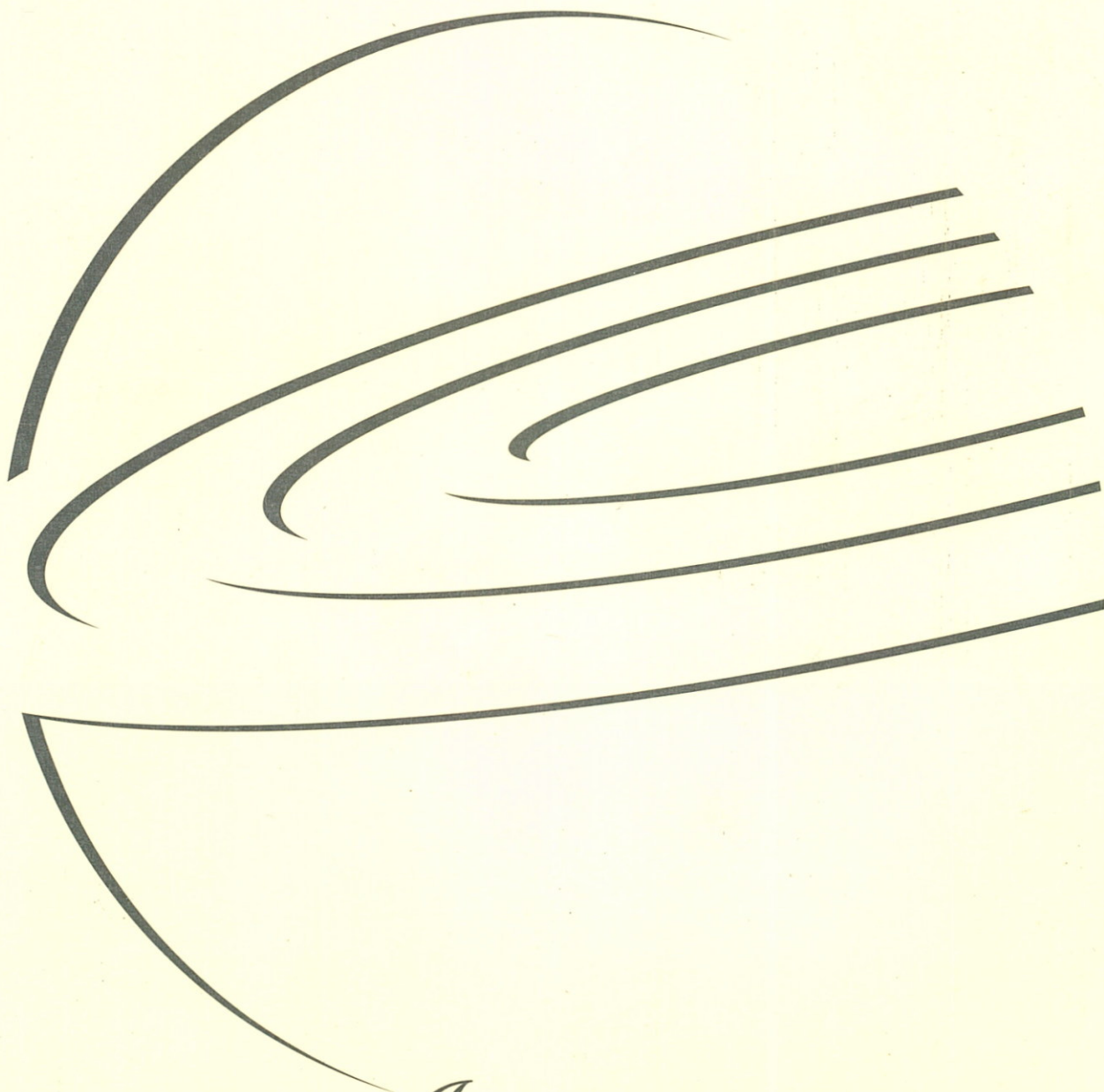


*Gábor Dénes Főiskola*

---

**Dr. Szalai Tibor**

**A MESTERSÉGES  
INTELLIGENCIA  
ALAPJAI**



*INOK*

Egy "kockás" teacher olyan  
eszköz amely a probléma  
specifikus ismeret megérte-  
sére képes és intelligencia  
kibontója a témakörrel  
ismertetés támogat egy  
feszített "működő"  
megoldástani aljainon  
feltevésekhez.



**Dr. Szalai Tibor**

# **A MESTERSÉGES INTELLIGENCIA ALAPJAI**

Nyitott rendszerű képzés – távoktatás –  
oktatási segédlet  
Felsőoktatási tankönyv

**INOK Kft.**  
Budapest, 2006

Szerkesztés, tördelés: **OFFICE-CAR Bt.**  
2040 Budaörs Csata u. 17.

**A jegyzet megrendelhető, illetve megvásárolható az  
LSI Informatikai Oktatóközpontban  
1037 Budapest, Bécsi út 324.  
Telefon: 436-6520  
Fax: 436-6521**

***ISBN 963 9625 35 3***

Kiadó: **INOK Kft.**  
Felelős vezető: **INOK Kft. ügyvezetője**  
Témafelelős: **Flier István**  
Készült: **OLITON Kft.**  
1104 Bp. Kada u. 149.

# TARTALOMJEGYZÉK

ELŐSZÓ.....	3
<b>1. A MESTERSÉGES INTELLIGENCIA FOGALMA, ÁGENS SZEMLÉLET .....</b>	<b>5</b>
1.1. A MESTERSÉGES INTELLIGENCIA FOGALMA .....	8
1.2. ÁGENS SZEMLÉLET A MESTERSÉGES INTELLIGENCIÁBAN.....	12
1.3. A MESTERSÉGES INTELLIGENCIA TÖRTÉNETE, FŐBB EREDMÉNYEI .....	14
<b>2. FELADATREPREZENTÁCIÓ GRÁFOKKAL, KERESÉSI ALGORITMUSOK.....</b>	<b>17</b>
2.1. A NEMINFORMÁLT (AZAZ VAK) KERESÉS MÓDSZEREI.....	21
2.2. AZ INFORMÁLT KERESÉS MÓDSZEREI.....	29
2.2.1. A legjobb először keresés .....	31
2.2.2. Iteratív javító algoritmusok.....	37
2.3. PÉLDÁK .....	41
<b>3. TUDÁSREPREZENTÁCIÓ ÉS KÖVETKEZTETÉS .....</b>	<b>45</b>
3.1. AZ ÍTÉLETKALKULUS .....	53
3.2. PÉLDÁK .....	58
3.3. AZ ELSŐRENDŰ LOGIKA (PREDIKÁTUM KALKULUS) .....	62
3.3.1. Következtetési eljárások .....	65
3.3.2. Általánosított Modus Ponens .....	69
3.3.3. Általánosított Rezolúció .....	72
3.3.4. Példák.....	73
<b>4. A BIZONYTALAN TUDÁS KEZELÉSE.....</b>	<b>79</b>
4.1. FUZZY LOGIKA.....	79
4.1.1. Fuzzy halmazok .....	80
4.1.2. Fuzzy relációk.....	81
4.1.3. Fuzzy következtető rendszer.....	81
4.1.4. Fuzzy Logic Controller.....	83
4.1.5. Fuzzyfikálás .....	84
4.1.6. Következtető rendszer és a szabályalapú tudásbázis .....	84
4.1.7. Defuzzifikáció.....	85
4.2. A VALÓSZÍNŰSÉGI LOGIKA.....	87
4.3. A BIZONYOSSÁGI TÉNYEZŐK MÓDSZERE .....	89
4.4. FUZZY ALGEBRA ALAPJAI.....	90



4.5.	PÉLDÁK .....	97
<b>5.</b>	<b>A TUDÁSREPREZENTÁCIÓ ALKALMAZOTT MÓDSZEREI.....</b>	<b>107</b>
5.1.	A TUDÁSÁBRÁZOLÁS ÉS KÖVETKEZTETÉS MÓDSZEREI .....	108
5.2.	1.1.2. SZABÁLY-ALAPÚ REPREZENTÁCIÓ ÉS KÖVETKEZTETÉS .....	109
5.3.	ESET-ALAPÚ KÖVETKEZTETÉS .....	112
5.4.	SZAKÉRTŐ RENDSZEREK FEJLESZTÉSE.....	114
	5.4.1. Szakértő keretrendszerek .....	115
	5.4.2. Fejlődési tendenciák.....	118
<b>6.</b>	<b>A GÉPI TANULÁS .....</b>	<b>141</b>
6.1.	A DÖNTÉSI FA MÓDSZERE .....	143
6.2.	A LEGKÖZELEBBI SZOMSZÉD MÓDSZERE.....	148
6.3.	A NEURÁLIS HÁLÓK MÓDSZERE .....	151
	6.3.1. Neurális modellek .....	152
	6.3.2. Tanulás neurális hálókkal .....	154
	6.3.3. Back propagation .....	155
	6.3.4. Felhasználási területek.....	157
	6.3.5. Hibrid megközelítések .....	158
	6.3.6. Alkalmazás.....	167
<b>7.</b>	<b>FIZIKAI ÁGENSEK – ROBOTIKA .....</b>	<b>169</b>
7.1.	AZ IPARI ROBOTOK MOZGÁSTÖRVÉNYEI. A ROBOTKAROK KINEMATIKÁJA .....	171
7.2.	JELLEGZETES IPARI ROBOT ÉPÍTŐELEMELK.....	173
7.3.	MEGFOGÓ SZERKEZETEK, ROBOTKEZEK.....	186
7.4.	ROBOTOK PROGRAMOZÁSA.....	194
7.5.	ROBOTOK ALKALMAZÁSA .....	203
<b>8.</b>	<b>INTELLIGENS GYÁRTÁS .....</b>	<b>215</b>
8.1.	HOLONIKUS GYÁRTÁSI MODELL .....	218
8.2.	FRAKTÁL VÁLLALAT MODELL.....	222

## ELŐSZÓ

A jegyzetet Dr. Váncza József A mesterséges intelligencia alapjai tárgyhoz kidolgozott segédletei és előadásai alapján, a Gábor Dénes Főiskola által Phare támogatással kiadott gyakorlati jegyzet – Horváth, Szalay: A mesterséges intelligencia gyakorlati alkalmazásai felhasználásával –, és a tárgy 2001. és 2002. évi gyakorló előadójaként szerzett tapasztalataim és vázlataim segítségével készült. A jegyzet részben tartalmaz összefüggő, szerkesztett összefoglaló részeket legtöbbször a fejezetek elején, és az előadásokon is kivetített, részletes vázlat fóliaképeit, amelyek segítségével a számonkérésre kerülő teljes anyag követhető és elsajátítható. A feladatmegoldási készség fejlesztése illetve a tananyag megértésének mélyítése érdekében számos példa, feladatmegoldás és alkalmazás is került a jegyzetbe. A jegyzet megírásával fő célom az volt, hogy a korábban elkészült tankönyv, jegyzet és segédlet, valamint a különböző egyéb forrásból beszerezhető további dokumentumok kusza és nem teljes dzsungel helyett egyetlen dokumentumra támaszkodva is fel tudjon készülni a vizsgázni kívánó hallgató. Meglehet a vázlatos fogalmazás miatt a tömör kifejezések alaposabb odafigyelést igényelnek a megértéshez, de a vázlatok részletessége és a bemutatott példák segítségével a jegyzetben szereplő tananyag önállóan is elsajátítható. Mindezek ellenére minden olvasónak javaslom, hogy a jegyzetet az előadásokkal kiegészítve, a konzultációs lehetőségeket igénybevéve, és az ajánlott szakirodalmak kiegészítő információit szükség esetén átnézve fogjon hozzá a felkészüléshez a biztos siker érdekében.

Budapest, 2002.

A szerző



## JELMAGYARÁZAT



– súlyponti rész, definíció, figyelmesen olvasandó



– kérdés vagy feladat



– magyarázat, példa



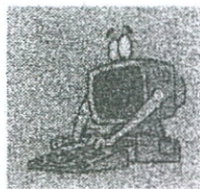
– írásban megoldandó feladat



– aprólékos, figyelmes áttanulmányozást igénylő rész, például levezetés, logikai vázlat, magyarázat



– kiegészítő ismereteket tartalmazó rész, olvasmány



– számítástechnikai – szoftver, illetve hardver – hivatkozások, alkalmazások



# 1. A MESTERSÉGES INTELLIGENCIA FOGALMA, ÁGENS SZEMLÉLET

A mesterséges intelligencia (*artificial intelligence, AI*) a számítástudomány azon területe, amely olyan módszerek és technikák kifejlesztésére törekszik, melyek révén a számítógépek intelligensnek tekinthető tevékenységeket tudnak végezni. Alapvető kérdés, hogy mi tekinthető intelligens viselkedésnek? Az intelligens jelző mindenképpen magába foglalja azt, “ami emberré tesz minket”, “amit az okos emberek tesznek”. Így az intelligens viselkedés kiterjed a hétköznapi tevékenységekre (pl. ismerős felismerése, fordítás, beszélgetés, autóvezetés) éppúgy, mint formális feladatok (pl. geometriai tétel bizonyítása, sakkozás) és szakértelmet igénylő feladatok (termék megtervezése, pénzügyi elemzés, orvosi diagnosztika) megoldására.

Miként érthető meg az intelligens viselkedés? Milyen struktúrák és eljárások rejlenek az intelligens viselkedés különféle megjelenései mögött? Ezekre és más hasonló problémákra csak kísérletezés nyomán várható válasz – így a mesterséges intelligencia kutatásban a tudományos és technikai problémák átfedik egymást. A felvetett kérdések a filozófia, pszichológia, nyelvészet, döntéelmélet, matematika területein is ismertek, de egyik sem vonja be úgy a számítógépet a kutatás módszerébe, mint a mesterséges intelligencia. Az emberi gondolkodás empirikus kutatásával foglalkozik a mesterséges intelligenciával szoros kapcsolatban álló kognitív pszichológia, amely modellezni próbálja mindazt, ami az emberi emlékezet működéséről, a problémamegoldás és tanulás mechanizmusairól, az emberi érzékelésről és a motoros viselkedésről tudható.

Ennél szerényebb, ám egyben többféle (nem csak emberi) viselkedésmintát megengedő célkitűzés az ún. racionális ágens létrehozása. A racionális ágens a rendelkezésre álló biztos ismeretekre és bizonytalan vélekedésekre támaszkodva a lehető legjobb cselekvést próbálja végrehajtani annak érdekében, hogy egy adott célt elérjen. A racionális ágens tehát optimál: saját várható hasznát maximalizálni próbálja a rendelkezésére álló eszközökkel. Az ágens (*agent*) megjelölés egy adott környezetben való cselekvésre helyezi a hangsúlyt, ahol a cselekvésbe beletartozik a környezet érzékelése éppúgy, mint a következtetés és a környezetbe való beavatkozás. Kifejezi, hogy a tudás, akármilyen mennyiségben és kifinomult formában legyen is megragadva, önmagában nem elegendő az intelligens viselkedéshez; azt mozgósítani is tudni kell, mégpedig adott célok elérése érdekében.

Bármely ágens egy hardver architektúrából és egy azon futó programból áll, ahol az architektúra megköti az érzékelés és a cselekvés lehetséges módzatait, míg a program érzékelésekhez akciókat rendel akár nagyon egyszerű, akár igen bonyolult módon. A mesterséges intelligencia feladata racionális ágensek programjainak tervezése és készítése. Ez a felfogás egyesítő, mert a racionális jelző a cselekvésre vonatkozik, és nem arra a módra, ahogyan az ágens eldönti, hogy mit tegyen. Egyszóval nem köti meg, hogy az ágens milyen belső mechanizmussal működjék. Az ágens-alapú megközelítés további előnye, hogy nyitott az elosztott intelligencia, a több ágenset megengedő (*multi-agent*) modell irányába, ahol a környezetet – egyebek között – más ágensek határozzák meg.



## ELŐADÁS

A mesterséges intelligencia alapjai

### Mi a mesterséges intelligencia?

#### Intelligencia

*“Sejtés, amivel felfedezünk valamely rejtett rendet.”*  
(H. Barlow, W.H. Calvin)

**mire szolgál?**

**nincs egyetlen formája**

**analitikus, kreatív, szociális, fizikai, ...**

**viszonylagos fogalom**

Dr. Szalay Tibor

3

A mesterséges intelligencia alapjai

### Mi tekinthető intelligens viselkedésnek?

**intelligens: "ami emberré tesz", "amit az okos emberek tesznek"**

- hétköznapi tevékenységek
- formális feladatok
- szakértő feladatok
- ???

Dr. Szalay Tibor

4



## 1.1. A mesterséges intelligencia fogalma

A mesterséges intelligencia alapjai

### Mi a célja (a mesterséges intelligenciának)?

*Emberi tevékenységek – akár tényleges, akár csak szándékolt – számítógép által való támogatása.*

### Miért éppen ez a megfogalmazás?

nem önmagára utaló

feladatra irányuló, ezáltal befogadó

az eredmény érzékelhető, mérhető, kiértékelhető

az emberi intelligencia kiterjesztése (nem utánzása,

**nem kiváltása)**

Dr. Szalay Tibor

5

A mesterséges intelligencia alapjai

### Definíció

**Artificial Intelligence - McCarthy 1956**

**Legelterjedtebb megközelítés (Russel 1996):**

	<b>emberhez hasonló</b>	<b>racionális</b>
<b>gondolkodnak</b>	(A) úgy gondolkoznak, mint az ember	(C) racionálisan gondolkoznak
<b>cselekszenek</b>	(B) úgy cselekszenek, mint az ember	(D) racionálisan cselekszenek

Dr. Szalay Tibor

6

## A) Emberhez hasonlóan gondolkodni

Az emberi gondolkodás megértése - pszichológia

A gondolkodás folyamatának leírása (programozása)

Fizikai megvalósítás

számítógépes programok

neurális hálózatok

genetikus algoritmusok

**Kognitív tudomány**

## B) Emberhez hasonlóan cselekedni

Turing teszt - Alan Turing

A számítógép intelligensnek tekinthető, amennyiben egy terminálon keresztül kérdező nem tudja eldönteni, hogy vele szemben ember, vagy számítógép áll.

Képességek:

- természetes nyelvmegértés
- tudásreprezentáció
- automatikus következtetés
- gépi tanulás



**Teljes Turing teszt**

- számítógépes látás                      - robotika

**Eliza** - Joseph Weizenbaum 1966 (mintaillesztés)

**K:** Az emberek mindig bámulnak engem.

**E:** Tudna mondani egy példát?

**K:** Például ön is bámul most engem.

**E:** Miért gondolja, hogy én bámulom önt?

**K:** Mert ön gyűlöl engem.

**E:** Tegyük föl, hogy gyűlölöm magát. Mit változtat ez a dolgokon?



**Megoldás mintaillesztéssel**

Séma	Válasz
1. X mindig Y	Tudna mondani egy példát? Mikor? Valóban mindig?
2. X ön Y engem Z	Miért gondolja, hogy én Y önt? Miért gondolja, hogy ön X én Y Z? Tegyük fel, hogy én Y magát Z. Mit változtat ez a dolgokon?

**Pszichológus - páciens párbeszéd (DOCTOR)**



## C) Racionálisan gondolkodni

Logika

Arisztotelész - szillogizmus

“Minden ember halandó” - “Szókratész ember”

=> “Szókratész halandó”

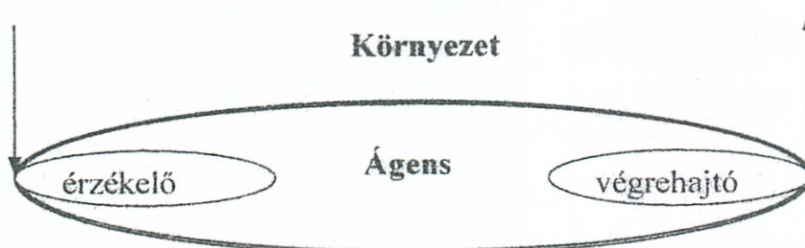
Modus ponens - következtetési szabály

## D) Racionálisan cselekedni

Az intelligencia mértéke a racionalitás (ideális nem emberi):

=> Adott helyzetben a legjobbat cselekedni

**Ágens** - érzékeli a környezetét és cselekszik



Felépítése:

Architektúra (hardware) + program (akciókat indít)

## 1.2. Ágens szemlélet a mesterséges intelligenciában

A mesterséges intelligencia alapjai

### Ágens definíció

Az **ágens** olyan rendszer, amely a következő **tulajdonságokkal** rendelkezik:

- **Beágyazottság** (környezetbe ágyazott)
- **Reaktivitás** (érezkeli környezetét és reagál az abban bekövetkezett változásokra)
- **Autonómia** (önálló működés)
- **Helyzetfüggőség** (csak helyzethez, szerephez kötötten léteznek)

Dr. Szalay Tibor

13

A mesterséges intelligencia alapjai

### Racionális ágens

Célja és információi vannak, és ezeknek megfelelő legjobb akciót választja. (legjobbat cselekszi - ideális intelligencia)

**További ágens tulajdonságok:**

- **Kezdeményezőkézség** (nemcsak reagál, de a cél érdekében beavatkozik)
- **Célvezérelt viselkedés**
- **Temporális kontinuitás** (huzamosabb ideig létezik)

Dr. Szalay Tibor

14



## Milyen lehet a környezet?

elérhető

(sakk - igen; orvosi diagnosztika, vezetés - nem)

determinisztikus

epizodikus

statikus/dinamikus

(sakk - statikus; sakk órával - félig dinamikus)

diszkrét/folytonos (játékok - diszkrét)

## Főbb ágens típusok I.

egyszerű reflex ágens

- feltétel  $\Rightarrow$  akció (ha  $\Rightarrow$  akkor) szabályok
- előre kiszámított reakciók, tanult válaszok (fékezés)

reflex ágens belső állapottal

- megkülönböztetés állapotok között, melyeket egyformának érzékeljük, de mégis mások
- miként változik a világ

## Főbb ágens típusok II.

### célirányos ágens

- cél = kívánt állapot leírása
- tervezés, előre gondolkodás
- hatékonyság, rugalmasság

### haszonelvű ágens

- mennyire hasznos (magának) amit tesz
- cél = előnyben részesített állapot
- változatok mérlegelése, célok közti választás

## 1.3. A mesterséges intelligencia története, főbb eredményei

## Az MI története I.



Arisztotelész Logika c. műve

Kopernikusz univerzum modellje (a valóság viselkedése és az észlelt jelenség szétválik)

Descartes Meditations c. munkája (fizikai valóság megkérdőjelezése "gondolkodom, tehát vagyok", test és lélek szétválasztása)

Euler a königsbergi hidak probléma kapcsán alkalmazza először a reprezentáció fogalmát



## Az MI története II.

Leibnitz az első formalizált logikai rendszer és automatikus számítások elvégzésére alkalmas gép

Charles Babbage az első mechanikus számítógép alkotója

Frege - az elsőrendű predikátumkalkulus kidolgozója  
(automatikus következtetések, tételbizonyítás)

Alan Turing - Turing teszt

Newell és Simon (60-as évek) General problem solver

...

## Úttörő alkalmazások

Szakértő rendszerek (MYCIN - vérfertőzéseket és agyhártyagyulladást diagnosztizálta)

Mesterséges látás (Rosenblatt perceptron modellje és a későbbi neurális hálózatok)

Természetes nyelvmegértés (Winograd SHRDLU rendszere az első)

Beszédértés

Gépi tanulás

Tervezés, robotika

A mesterséges intelligencia alapjai

### **MI alkalmazások**

10.000 – 20 000 működő szakértő rendszer, pl.

- számítógép ipar (COMPAQ, MICROSOFT)
- háztartás (Whirlpool)
- pénzügyek (AMEX, Swiss Bank)
- légitársaságok (Air France, KLM, AA)
- gyártásütemezés

komplex cselekvéstervező rendszerek

- DART (Sivatagi Pajzs hadművelet tervezése)
- Remote Agent Experiment (RAX)
- Deep Space One (1999)

**Adatbányászat**



## 2. FELADATREPREZENTÁCIÓ GRÁFOKKAL, KERESÉSI ALGORITMUSOK

A mesterséges intelligenciában a problémák egyik jellegzetes tulajdonsága, hogy a megoldásukhoz próbálkozásokon, kereséseken keresztül juthatunk el. A feladatok jelentős része reprezentálható egy irányított gráffal, így a mesterséges intelligencia egyik alapvetően fontos területe a gráfokon történő keresés.

A mesterséges intelligencia korai kutatási területeinek egyike a gráfkeresési algoritmusok. Ebben a fejezetben a gráffal történő feladatrepresentációt mutatjuk be, szó lesz az informálatlan és az informált keresési algoritmusokról, és azokról a heurisztikus keresésekről is, amelyek nem építenek gráfot, úgy jutnak el a megoldáshoz.

A gráffal történő feladatrepresentáció esetén a feladat állapotterét írjuk le egy irányított gráf segítségével. A gráf csomópontjai a különböző állapotokat jelölik, a gráf éleinek pedig az állapotok változását előidéző operációk felelnek meg. Kiemelt fontossága van a kiinduló állapotnak és a célállapotnak, amelyekből több is lehetséges. Egy feladat megoldásának azt a műveletsort (operátor sorozatot) tekintjük, amely a kezdőállapotból elvezet a célállapotba. A gráfrepresentáció során ezt a megoldást megoldási útnak is nevezzük. Több ilyen út esetén a keresés célja a legrövidebb út, avagy a legkisebb költséggel járó műveletsor megtalálása.

# ELŐADÁS

A mesterséges intelligencia alapjai

## Feladat

### 8-as játék

2	8	3
1	6	4
7		5

kezdeti állapot

1	2	3
8		4
7	6	5

cél állapot

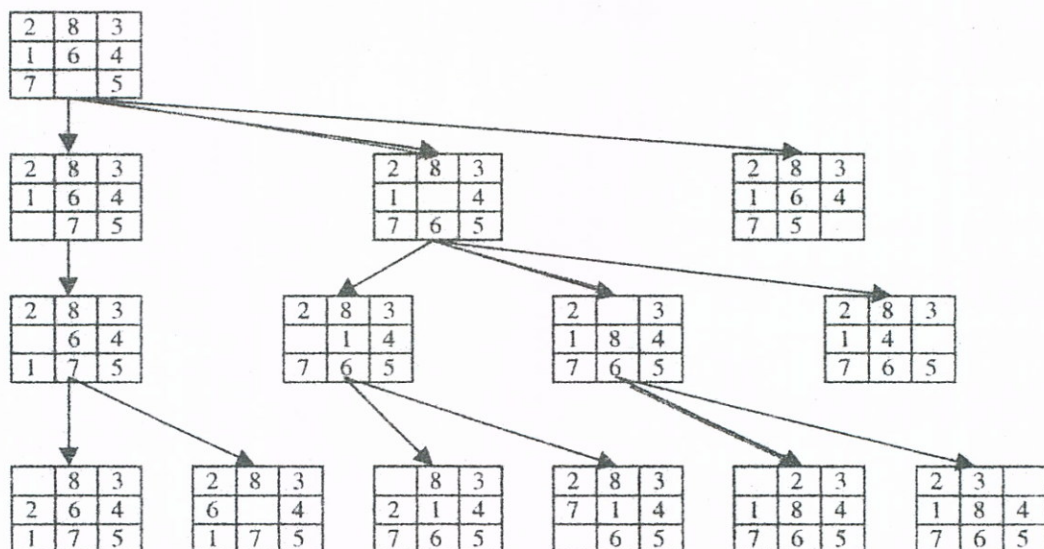
Hogyan fogalmazzuk meg a játék menetét, illetve a célállapot megközelítését, elérését? - Reprezentáció

Dr. Szalay Tibor

22

A mesterséges intelligencia alapjai

### Egy lehetséges reprezentáció



Dr. Szalay Tibor

23



A mesterséges intelligencia alapjai

## Probléma

mit kell tudni egy feladat megoldásához?

hogyan kell az ismereteket reprezentálni?

hogyan kell megoldást keresni?

Az intelligencia biztos jegye: hogyan használjuk tudásunkat a keresési problémák egyszerűsítésére.

Témakörök

vak (informálatlan) keresés

heurisztikus keresés

Dr. Szalay Tibor

34

A mesterséges intelligencia alapjai

## Alapfogalmak

### Keresési probléma

kezdeti állapot (i),  
operátorok,  
cél tesz (g),  
költségek

### A keresés folyamata

állapottér, út, megoldás  
kiterjesztés, stratégia

### A keresés technikája

keresési fa, csomópont, gyökér, levél  
keresési algoritmus: melyik csomópontot érdemes kiterjeszteni?  
a fa jellemzői: - mélység (d),  
- elágazási tényező (b)

A keresés költsége      út költsége + keresési eljárás költsége

Dr. Szalay Tibor

35

A mesterséges intelligencia alapjai

### 8-as játékra alkalmazva

2	8	3
1	6	4
7		5

kezdeti állapot(i)

1	2	3
8		4
7	6	5

cél állapot(g)

Állapotok: számozott lapok helyzete (üres mező helyzete)

Operátorok: az üres mező mozgatása jobbra, fel, balra, le

Költség: műveletek (lépések) száma

Megoldás: minimális számú lépés a kezdeti és a végállapot között

Dr. Szalay Tibor

26

A mesterséges intelligencia alapjai

## Általános keresési algoritmus

Adott: kezdeti állapot, cél teszt és az operátorok

Általános keresés

- 1) Legyen L a kezdeti állapotokat tartalmazó lista.
- 2) Ha L üres, akkor állj le, a keresés sikertelen; egyébként legyen n egy csomópont L-ből.
- 3) Ha n egy célállapot, akkor állj le és add vissza (a hozzá vezető úttal együtt) eredményként.
- 4) Egyébként töröld n-et L-ből;  
 állítsd elő n gyermekeit;  
 jegyezd fel a hozzájuk vezető utat;  
 add a gyermekeket L-hez;  
 menj vissza 2)-re.

Dr. Szalay Tibor

27



A mesterséges intelligencia alapjai

### Legfontosabb döntés

Hogyan választunk csomópontot L-ből (2)?  
 (vak keresés - heurisztikus (informált) keresés)

### Keresési stratégiák tulajdonságai

- teljesség
- időbeli komplexitás
- memória komplexitás
- optimalitás

Dr. Szalay Tibor

28

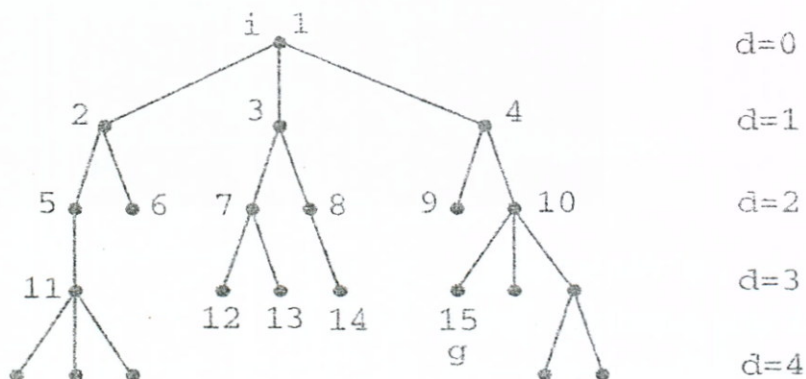
## 2.1. A neminformált (azaz vak) keresés módszerei

A mesterséges intelligencia alapjai

### Neminformált (vak) keresések

#### Szélességben először keresés

Mindig a legmagasabb szinten levő csomópontot terjesztjük ki



Dr. Szalay Tibor

29

### Tulajdonságok

teljes

optimális

időbeli komplexitás  $\approx 1 + b + b^2 + \dots + b^d \approx b^d$

memória komplexitás  $\approx b^d$

mélység	csomópontok	idő	memória
4	11.111	11 sec	1 MB
6	106	18 min	111 MB
8	108	31 óra	11 GigaB
10	1010	128 nap	1 TeraB
12	1012	35 év	111 TeraB

( $b=10$ , 1000 csomópont/sec, 100 byte/csomópont)

### Egységes költségű keresés

A keresési fában mindig a pillanatnyilag legkisebb költségű csomópontot terjeszti ki. (Ha a költség,  $g(n) = d(n)$ , vagyis ha minden él költsége egységnyi, akkor szélességben először keresés.)

### Tulajdonságok

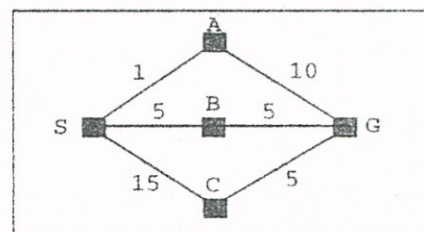
teljes

optimális (feltételesen)

időbeli komplexitás:  $b^d$

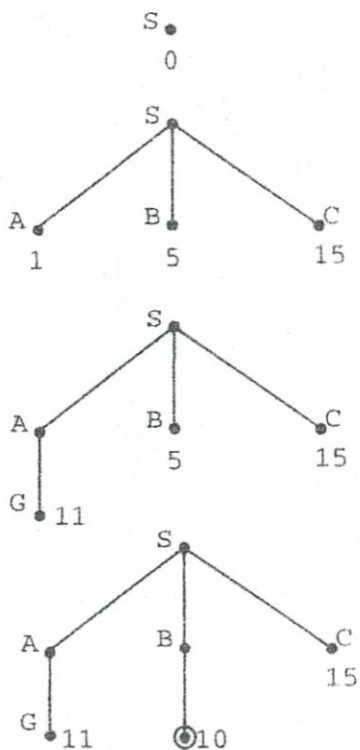
memória komplexitás:  $b^d$

### Példa





A mesterséges intelligencia alapjai



**Megoldás**

A megtalált első "suboptimális megoldásnál még van jobb, ezért még érdemes késleltetni az elfogadást!

Dr. Szalay Tibor

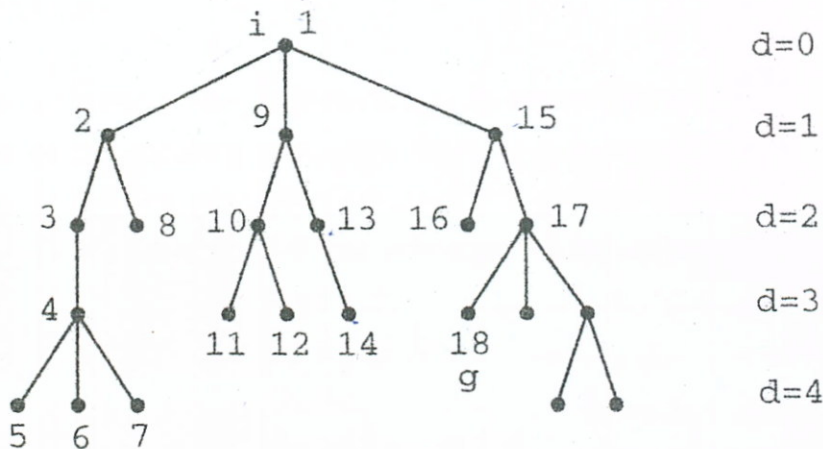
32

A mesterséges intelligencia alapjai

**Mélységben először keresés**

A keresési fában mindig a legmélyebben lévő csomópontok valamelyikét terjeszti ki.

**Példa:**



Dr. Szalay Tibor

33

### Tulajdonságok

memória komplexitás:  $b \cdot d$  (nagyon szerény igény!)

időbeli komplexitás:  $b^d$

nem teljes! (elakadhat egy rossz - esetleg végtelen - ágon)

nem optimális

### Korlátozott mélységű keresés

Mélységben először való keresés mélységi korláttal.

A keresési fában mindig a legmélyebben lévő csomópontok valamelyikét terjeszti ki, feltéve, hogy az nincs egy előre adott mélységi korlát ( $l$ ) alatt.

A korlát lehet önkényes, de eredhet a feladat természetéből.

### Tulajdonságok

memória komplexitás:  $b \cdot l$

időbeli komplexitás:  $b^l$

teljes (ha  $l$  nagyobb, mint a megoldás mélysége,  $d$ )

nem optimális



## Iteratív mélyítés

Megkerüli a mélységi korlát meghatározását.  
Korlátozott mélységű keresés egyre növekvő  $l = 0, 1, 2, \dots$  mélységi korlát mellett.

Mintha a csomópontoknak a mélységi korlát alatt nem lehetnének leszármazottai. Ha a célt nem sikerült így elérni, eggyel növeli a korlátot és újraindítja az egész keresést.

## Algoritmus (Iteratív mélyítés)

- 1) Legyen a mélységi korlát  $l = 0$ .
- 2) Legyen  $L$  a kezdeti állapotokat tartalmazó lista.
- 3) Legyen  $n$  az első csomópontot  $L$ -ből. Ha  $L$  üres, akkor növeld eggyel  $l$  értékét, és menj vissza 2)-re.
- 4) Ha  $n$  egy célállapot, akkor állj le és add vissza (a hozzá vezető úttal együtt) eredményként.
- 5) Egyébként töröld  $n$ -et  $L$ -ből;  
ha  $n$  mélysége kisebb mint  $l$ , akkor állítsd elő  $n$  gyermekeit;  
jegyezd fel a hozzájuk vezető utat;  
add a gyermekeket  $L$  elejéhez;  
menj vissza 3)-ra.

limit=0



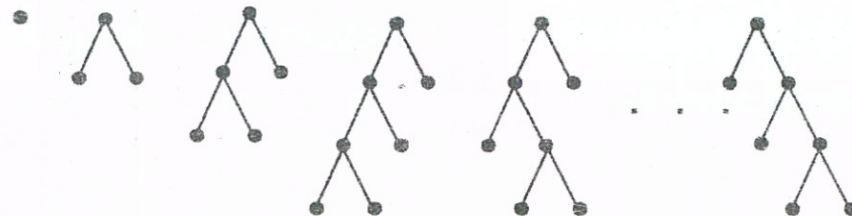
limit=1



limit=2



limit=3



## Szemléltetés

## Előnyei

szélességben-először keresés teljessége

mélységben először keresés csekély memória komplexitása

biztosítja, hogy a keresési fa nem lesz mélyebben feltárva,  
mint amilyen mélyen maga a cél található

## Alkalmazható

ha nagy a keresési tér és ismeretlen a megoldás mélysége

két személyes játékok

időkorlátos keresési feladatok



## Gráfkeresés

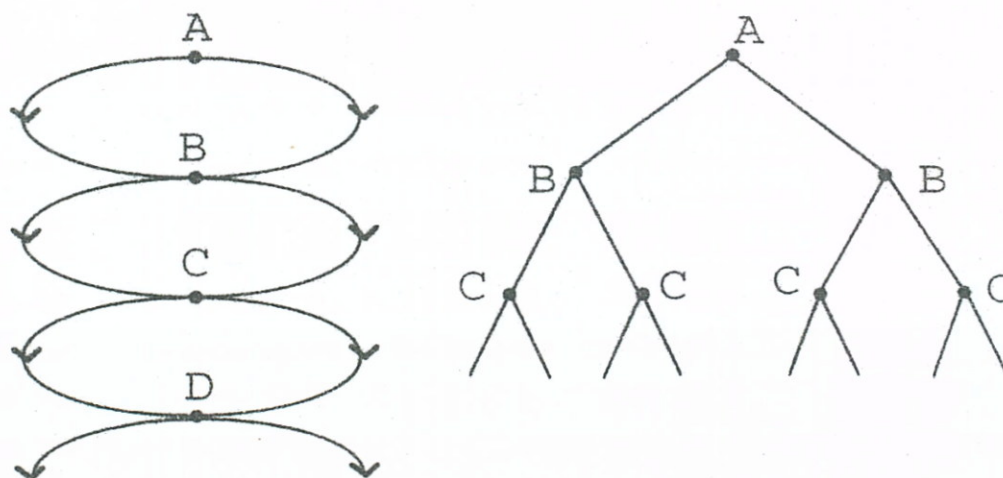
Egyazon csomóponthoz több út is vezethet. (pl. útkeresés)

A keresés hatékonysága javítható, ha sikerül felfedezni, hogy egy állapotba több út is vezet.

Ismétlődő állapotok kezelése: kiterjesztés tiltva  
 a szülő felé  
 az elődök felé  
 az összes, már kiterjesztett csomópont felé.

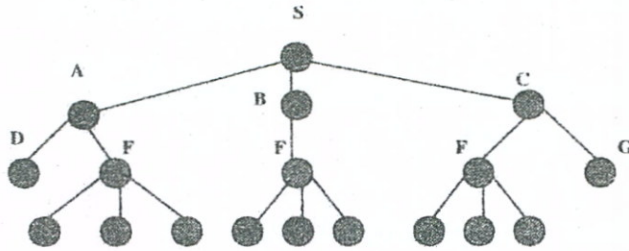
Megéri a széleskörű ellenőrzés? Ha sok a hurok az állapottérben, akkor igen.

## Példa

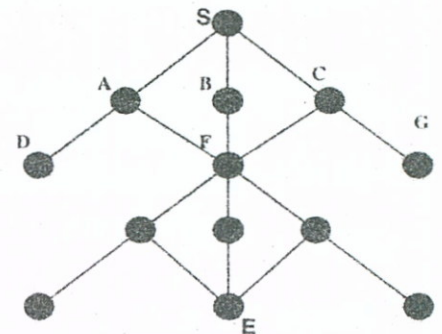


A mesterséges intelligencia alapjai

A keresési fa



Ha a gráf tartalmaz hurkokat is



Dr. Szalay Tibor

3

A mesterséges intelligencia alapjai

Összehasonlítás

	teljesség	idő	memória	optimalitás
SzE	igen	bd	bd	igen <sup>1</sup>
EK	igen	bd	bd	igen
ME	nem	bm	b*m	nem
KM	ha d = l	bl	b*l	nem
IM	igen	bd	b*d	igen

1) Csak ha minden ág azonos költségű a keresési fában.

Dr. Szalay Tibor

43



## 2.2. Az informált keresés módszerei

A mesterséges intelligencia alapjai

### Heurisztikus keresés

Hogyan javítható a vak keresési eljárások hatékonysága?

Hogyan lehet a feladatok megoldásával járó számításigényt csökkenteni?

Hogyan lehet keresni gyakorlatban korlátozott erőforrások alkalmazás mellett?

Hogyan lehet az adott területtel kapcsolatos információkat figyelembe venni, segítségül hívni?

Dr. Szalay Tibor

8

A mesterséges intelligencia alapjai

### Heurisztika

1) általános jelentés: bármely tanács, mely gyakran hatékony, ám nem biztos hogy minden esetben érvényes

2) technikai jelentés: heurisztikus kiértékelő függvény, mely a probléma egy állapotához egy számot rendel (pl. sakk: egy pozíció ere-je)

Dr. Szalay Tibor

9

A mesterséges intelligencia alapjai

### Heurisztika a keresésben:

A még ki nem terjesztett csomópontok (L) kiértékelése:  
mennyire van közel a célhoz?

### Algoritmus típusok

- legjobbat-először keresés: a legjobbnak tűnő csomópont kiterjesztése (mohó keresés, A\*, IDA\*)
- iteratív javítás: elmozdulás a legjobbnak tűnő irányba (csúcsra-mászás, szimulált hűtés, genetikus alg.)

Ám a legjobb csomópont megtalálása elvben nem könnyebb, mint maga a keresési feladat!

Dr. Szalay Tibor

10

A mesterséges intelligencia alapjai

### Heurisztikus kiértékelő függvény, $f(n)$

#### Célja:

- A) a feladat megoldásával járó számítási igény csökkentése
- B) adott erőforrás használat mellett a lehető legjobb megoldás megtalálása

Kompromisszum a számítási igény és a megoldás minősége között

Már kevés alkalmazás-függő szakismeret is nagyban segíthet akár A), akár B) célok érdekében.

Dr. Szalay Tibor

11



A mesterséges intelligencia alapjai

## Milyen legyen a kiértékelő függvény?

hatékonyan számítható

lehetőleg ne becsülje túl a tényleges költséget

### Problémák:

a kiértékelés lehet igen drága (idő, memória)

csak becslésen alapulhat (így félre is vezethet)

Dr. Szalay Tibor

12

### 2.2.1. A legjobbat először keresés

A mesterséges intelligencia alapjai

## Legjobbat először keresés: mohó keresés

Indíték: megtalálni egy célt, amilyen gyorsan csak lehetséges

Kiértékelés alapja: egyedül a céltól való távolság

Mindig azt a csomópontot terjeszti ki, amelynek minimális a céltól való távolság-becslése.

Dr. Szalay Tibor

13



A mesterséges intelligencia alapjai

### Algoritmus (Mohó keresés)

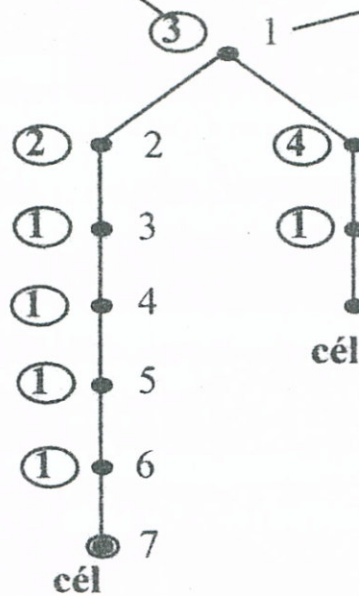
- 1) Legyen L a kezdeti állapotokat tartalmazó lista.
- 2) Ha L üres, akkor állj le - a keresés sikertelen; egyébként legyen n az a csomópont L-ből, amelyik várhatóan a legközelebb van a célhoz.
- 3) Ha n egy célállapot, akkor állj le és add vissza (a hozzá vezető úttal együtt) eredményként.
- 4) Egyébként töröld n-et L-ből; állítsd elő n gyermekeit; jegyezd fel a hozzájuk vezető utat; add a gyermekeket L-hez; menj vissza 2)-re.

Dr. Szalay Tibor

14

A mesterséges intelligencia alapjai

becsült távolság a céltől kiterjesztés sorrendje



### Példa

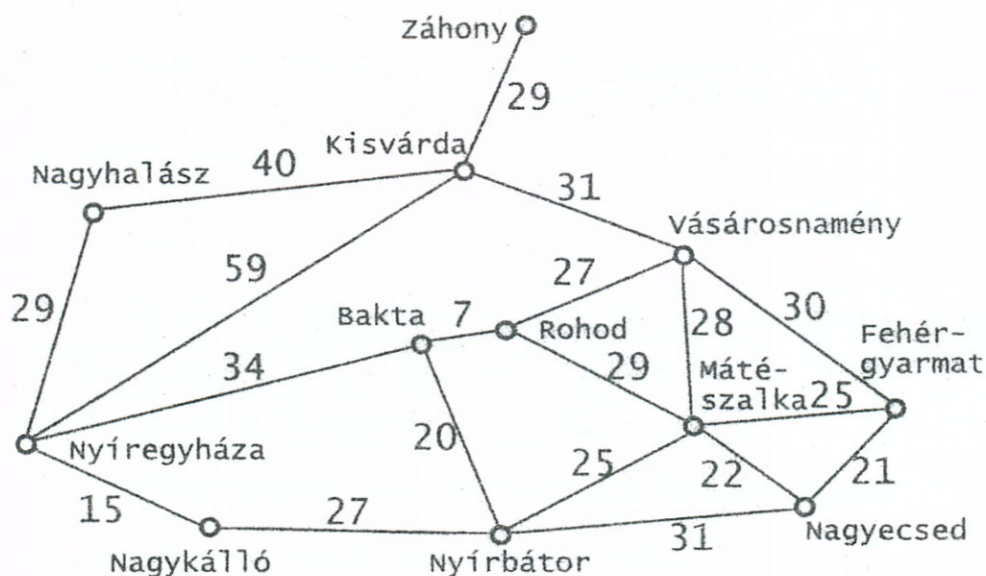
Nem az optimális célhoz vezető utat találja meg!

Dr. Szalay Tibor

15



## Feladat - Útkeresés



Dr. Szalay Tibor

16

## Útkeresés városok közt (gráfon)

kezdeti állapot: Nyíregyházán vagyunk

operátorok: utazás szomszédos városok között

állapotok: egyes városokban vagyunk

cél teszt: Fehérgyarmaton vagyunk már?

költség: megtett út hossza

optimális megoldás: minimális költségű út a kezdeti és a célállapot között

Dr. Szalay Tibor

17

A mesterséges intelligencia alapjai

## Megoldás mohó kereséssel

Legyen a heurisztikus függvény:

$$f(n) = \text{légvonalbeli távolság a céltól}$$

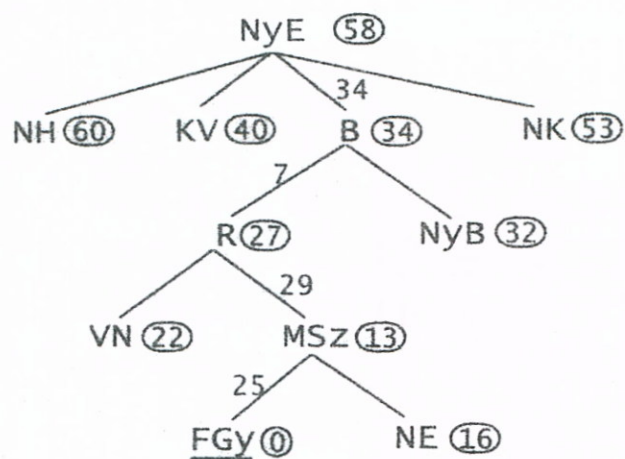
távolság légvonalban Fehérgyarmattól										
NyE	NH	KV	Z	NK	VN	R	B	NyB	MSz	NE
58	60	40	52	53	22	27	34	32	13	16

Dr. Szalay Tibor

18

A mesterséges intelligencia alapjai

## Keresési fa



teljes úthossz:  $34+7+29+25=95$   
nem optimális!

Dr. Szalay Tibor

19



### Legjobbat-először keresés: A\* keresés

A minimális útköltségű megoldást szeretnénk megtalálni.

Kiértékelés alapja:

- 1) a már megtett út, és
- 2) a még várható út költsége

$$f(n) = g(n) + h(n) \text{ ahol}$$

$g(n)$ :  $n$  tényleges távolsága a kezdeti állapottól

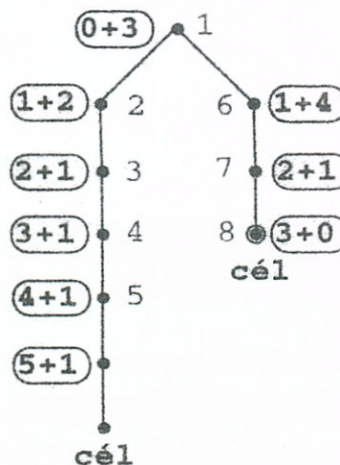
$h(n)$ :  $n$  becsült távolsága a céltól

### A\* algoritmus

Mindig azt a csomópontot válasszuk, amelyikre

$$f(n) = g(n) + h(n) \text{ minimális}$$

**Példa:**



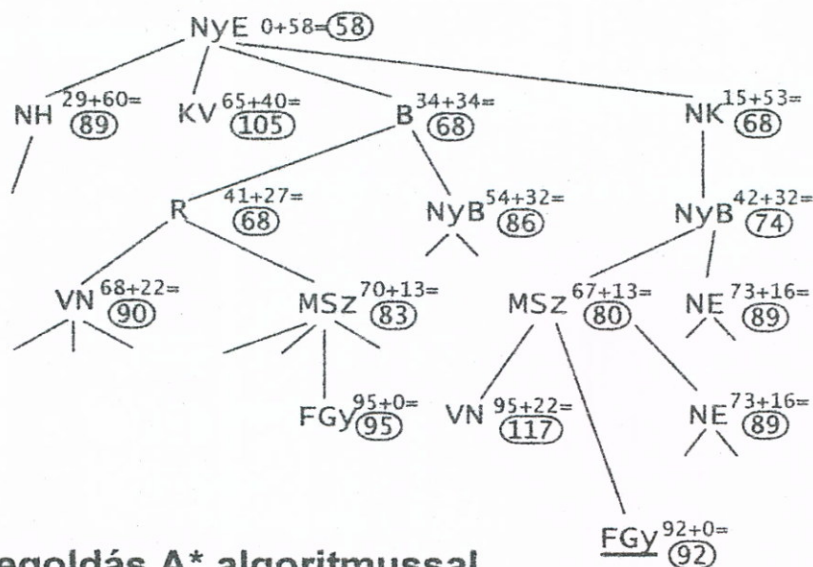
## Az A\* tulajdonságai

**teljes:** Ha az elágazási tényező (b) véges, és minden operátor pozitív költséggel jár

**optimális és teljes:** Ha minden n csomópontra igaz, hogy  $h(n)$  nem becsüli túl n tényleges távolságát a céltól, akkor A\* soha sem ad vissza optimálistól eltérő megoldást.

**komplexitás:** A kiterjesztett (tárolt) csomópontok száma még mindig exponenciálisan függ a megoldás méretétől.

Nagy problémák megoldásakor általában előbb fogy ki a memóriából, mint az időből



### Megoldás A\* algoritmussal

legrövidebb út: NYE-NK-NyB-MSz-Fgy

teljes úthossz:  $15+27+25+25=92$

optimális



## 2.2.2. Iteratív javító algoritmusok

A mesterséges intelligencia alapjai

### Iteratív javító algoritmusok

Probléma típus: megoldás a csomópontokban  
(pl. elrendezés tervezés - áramkör, gyártóberendezés)

Alapötlet: analógia a felületen való tájékozódással és mozgással

magasság: csomópont jósága (heurisztikus kiértékelés)

csúcsok: optimális megoldások

tájékozódás: adott pont szomszédságában lévő pontok kiértékelése

#### Algoritmusok

csúcsra mászás (*hill-climbing*)

szimulált hűtés (*simulated annealing*)

tabu keresés (*tabu search*)

genetikus algoritmusok

Dr. Szalay T. Horvát

24



A mesterséges intelligencia alapjai

### Genetikus algoritmusok

Alapötlet: Keresés a természetes kiválasztódás  
(durva) utánpótlásával.

#### Fogalmak

populáció

rátermettség (fitness)

genetikus kód

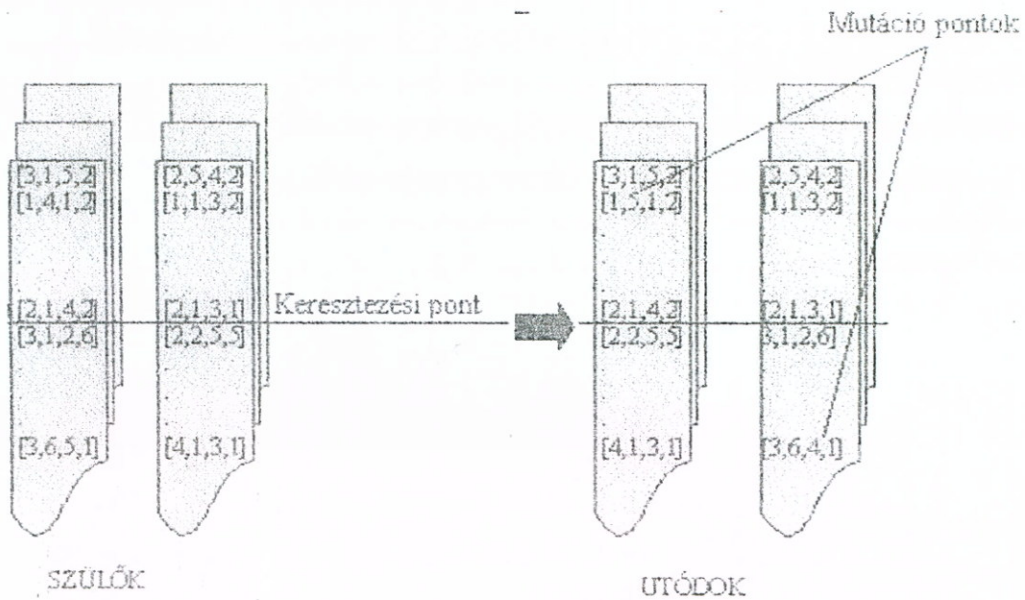
genetikus műveletek: kereszteződés, mutáció

Dr. Szalay T. Horvát

25

A mesterséges intelligencia alapjai

## Genetikus operátorok



Dr. Szalay Tibor

26

A mesterséges intelligencia alapjai

## Genetikus algoritmus

- 1) Töltsd fel a kezdeti populációt.
- 2) Ha a leállási feltétel teljesül, akkor állj le és add vissza a legjobb egyedeket eredmény-ként.
- 3) Egyébként



válassz ki néhány egyed a populációból;  
 alkalmazd rájuk a genetikus műveleteket;  
 értékeld ki az új egyedeket;  
 dobd el a legrosszabb egyedeket;  
 menj vissza 2)-re.

Dr. Szalay Tibor

27



## Csúcsra-mászás

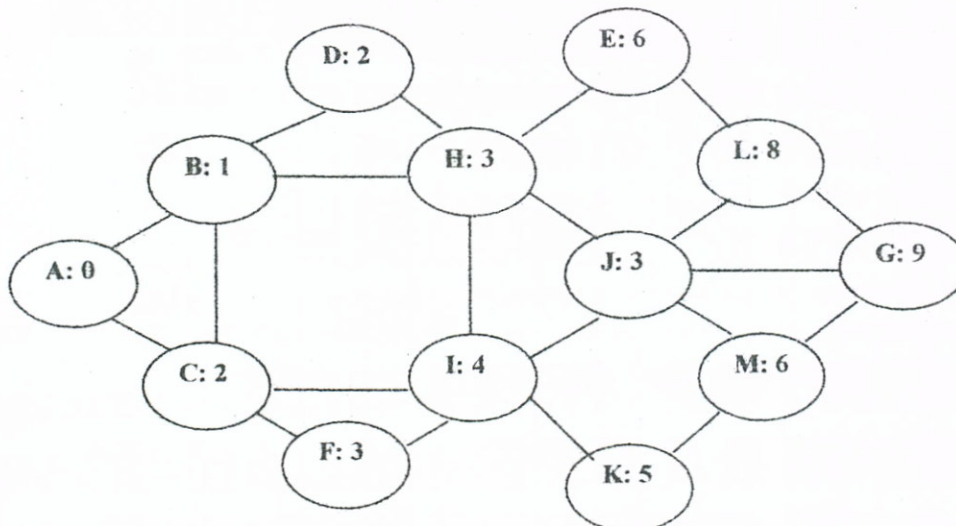
A keresés során egy csomópont

- 1) közvetlen leszármazottait vizsgálja csak, és
- 2) ezek közül mindig a legjobbat választja

### Algoritmus

- 1) Legyen  $n$  a kezdeti állapot.
- 2) Ha  $n$  egy célállapot, akkor állj le és add vissza eredményként.
- 3) Egyébként  
 állítsd elő  $n$  valamennyi  $n'$  leszármazottját;  
 legyen  $n = a$  legjobb  $n'$ ;  
 menj vissza 2)-re.

## Példa



A mesterséges intelligencia alapjai

## Tulajdonságok

Nem tárolja a keresési gráfot, csak a pillanatnyilag vizsgált csomópontot - így minimális memória igény

Sikere nagyban függ a felület alakjától

## Problémák

- lokális magaslat
- kiterjedt síkvidék
- gerinctúra lenne jó, de arra nem vezet út

Dr. Szalay Tibor

30

A mesterséges intelligencia alapjai

## Szimulált hűtés

Alapötlet: Fémöntési technikával való analógia

### Algoritmus

nem a legjobb lépést választja, hanem véletlenül választ  
 ha a lépés javít a pillanatnyi helyzeten, elfogadja és megteszi  
 bizonyos valószínűséggel elfogad olyan lépést is, amely ront a pillanatnyi helyzeten

hőmérséklet: befolyásolja a rontó lépések elfogadásának valószínűségét  
 - kisebb hőmérsékleten kisebb az esély  
 - a keresés előrehaladtával a hőmérséklet csökken (végül csúcsra-mászás)

Ha a hőmérséklet lassan csökken, megtalálja a globális optimumot.

Viszonylag új módszer, ám már meglepően sikeres alkalmazásokkal.

Dr. Szalay Tibor

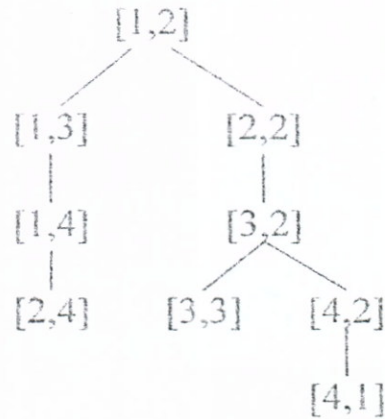
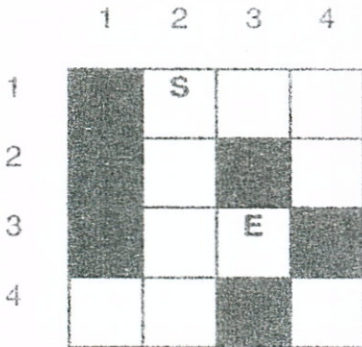
31



### 2.3. Példák

A mesterséges intelligencia alapjai

#### Feladat reprezentáció



Operátorok:

Lépés fel, jobbra, le, balra

A lépések költsége 1

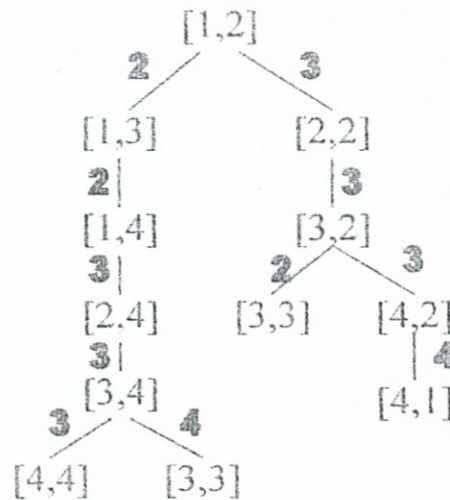
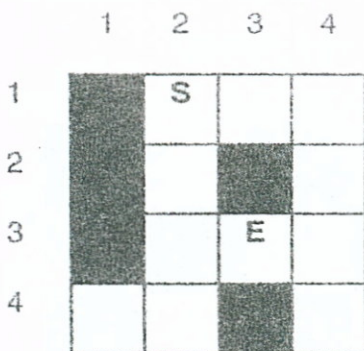
Ez gyakorlatilag mélységben először keresés

Dr. Szalay Tibor

32

A mesterséges intelligencia alapjai

#### Ha a lépések költsége nem egyenletes



Operátorok:

Lépés fel, jobbra, le, balra

Költsége 1 2 3 4

Mohó keresés

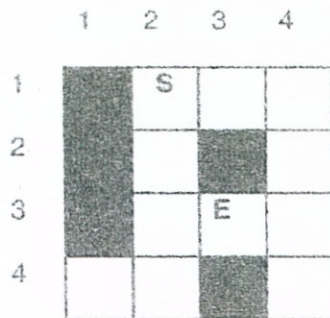
Dr. Szalay Tibor

33

A mesterséges intelligencia alapjai

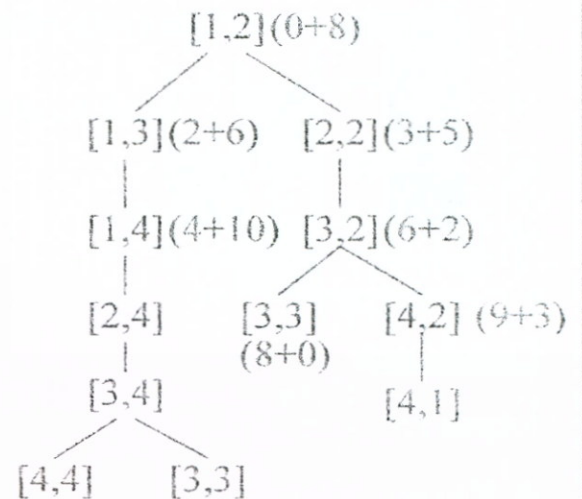
**A\* algoritmussal**  $f(n) = g(n) + h(n)$

$$h(n) = [\text{sgn}(x(E)-x(n))+2] * \text{abs}(x(E)-x(n)) + [\text{sgn}(y(E)-y(n))+3] * \text{abs}(y(E)-y(n))$$



Operátorok:

Lépés	fel,	jobbra,	le,	balra
Költsége	1	2	3	4



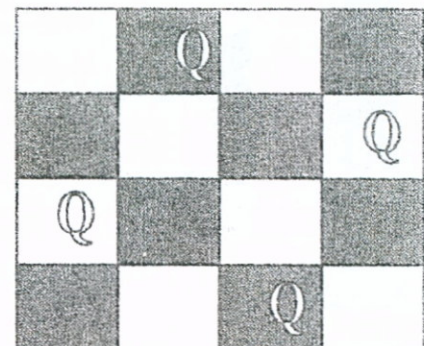
Dr. Szalay Tibor

34

A mesterséges intelligencia alapjai

### Reprezentáció - 4 királynő

Adott a 4 királynő probléma egy célállapota. (A 4 királynő olyan elhelyezése egy 4x4-es sakk táblán, hogy ne üthessék egymást.) A feladat a célállapotnak megfelelően felhelyezni a bábúkat az üres táblára.



Milyen reprezentáció? (állapotok, operátorok)

Dr. Szalay Tibor

35



1. reprezentáció

állapotok: 0-8 királynő bárhol a táblán

operátorok: királynő felhelyezése szabad mezőre

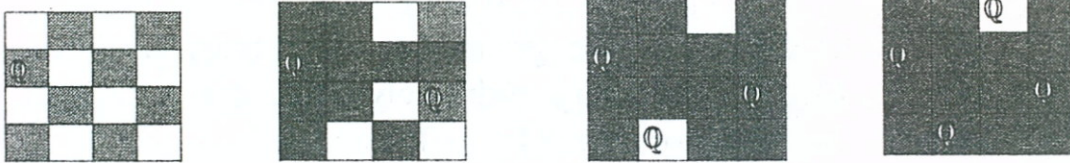
- 1. lépés (szint): 16 lehetséges állapot (elágazás)
- 2. lépés (szint): maradt 15 lehetséges állapot (elágazás)
- 3. lépés (szint): maradt 14 lehetséges állapot (elágazás)
- 4. lépés (szint): maradt 13 lehetséges állapot (elágazás)

47296

2. reprezentáció

állapotok: 0-8 királynő felhelyezése, hogy ne  
üssék egymást

operátorok: királynő felhelyezése szabad mezőre



- 1. lépés (szint): 16 lehetséges állapot (elágazás)
- 2. lépés (szint): maradt 6 lehetséges állapot (elágazás)
- 3. lépés (szint): maradt 2 lehetséges állapot (elágazás)
- 4. lépés (szint): maradt 1 lehetséges állapot (elágazás)

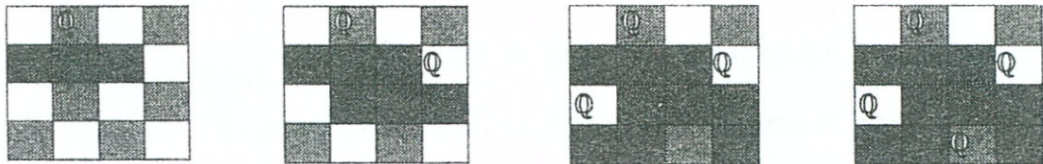
336

A mesterséges intelligencia alapjai

### 3. reprezentáció

állapotok: soronként egy-egy királynő a táblán

operátorok: 1-1 királynő új mezőre helyezése saját sorában, hogy ne üsse az addig felhelyezetteket



- 1. lépés: 4 lehetséges állapot
- 2. lépés: 1 lehetséges állapot
- 3. lépés: 1 lehetséges állapot
- 4. lépés: 1 lehetséges állapot

} 16

Dr. Szalay Tibor

38

A mesterséges intelligencia alapjai

### Hanoi tornyai

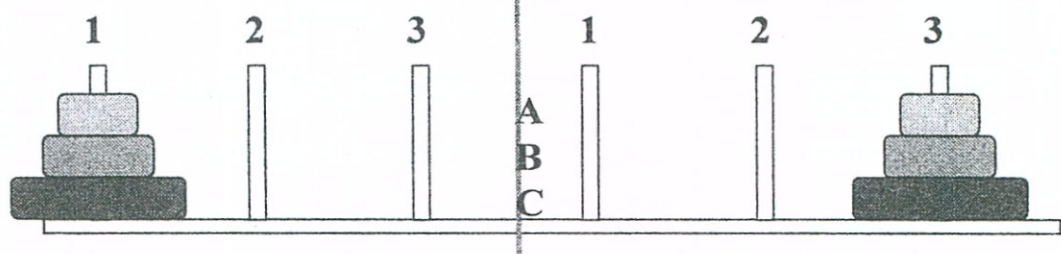
Állapottér reprezentáció:

$v \in \text{Vektor} ([A, B, C]; \{1, 2, 3\})$

Kezdőállapot: [1, 1, 1]      Célállapot: [3, 3, 3]

Feltétel: kisebb átmérőjű korongra nem kerülhet nagyobb

Operátorok: Mij az I. rúdról a j. rúdra helyezük át a legfelső korongot ( $1 \leq i, j \leq 3$ )



Dr. Szalay Tibor

39



### 3. TUDÁSREPREZENTÁCIÓ ÉS KÖVETKEZTETÉS

A mesterséges intelligencia kutatás mindmáig meghatározó (sokáig kizárólagos) feltevése, hogy a szimbólumokkal eredményesen bánni tudó gondolkodás az intelligencia előfeltétele, hogy minden intelligens tevékenység szimbólumok manipulálására - azok létrehozására, módosítására, kombinálására, és törlésére - vezethető vissza. Ahhoz, hogy az intelligens viselkedést rekonstruáljuk, nem kell mást tenni, mint egy másik fizikai médián - ez esetben a számítógépen - olyan rendszert létrehozni, amelyben mind a szimbólumok, mind az azokat manipuláló eljárások definiálhatók, és az eljárások működésbe is hozhatók. További megszorítás, ha megkívánjuk, hogy a szimbólumrendszer megalapozott legyen: azaz a szimbólumokat meg lehessen feleltetni a környezet objektumainak. A deklaratív álláspont sarkköve az a hipotézis, hogy a fizikai médián megvalósított, megalapozott szimbólumrendszer az intelligencia szükséges és elégséges feltétele. Ez a hipotézis vitatott: kétségbe vonható, hogy kell-e egyáltalán reprezentáció és következtetés az intelligens viselkedés hátteréül, szükség van-e szimbólumokra, illetve arra, hogy azok megalapozottak legyenek. A racionális ágens modelljében az esetleges szimbolikus eszközökkel megvalósított kognitív képességek (pl. következtetés) annak szolgálatában állnak, hogy az ágens megtalálja a helyes cselekvést. Jó alapot adnak a racionális viselkedés megvalósításához, de annak nem feltétlen és egyedüli előfeltételei. Azaz a racionális ágens fogalma egyesítő: magába foglalja mind a szimbolikus, mind például a neurális háló alkalmazásán alapuló módszerek használatát.

A tudásábrázolás (vagy -reprezentáció), és a hozzá elválaszthatatlanul kapcsolódó következtetés a mesterséges intelligencia egyik alapkérdése: miként modellezhető a világ úgy, hogy a modell elégséges alapot adjon a racionális cselekvéshez? Maga a tudásreprezentáció lehet a világ valódi dolgait helyettesítő pótlék, amely lehetővé teszi, hogy következtetéseket vonjunk le a világról, s ne csak tevékenykedjünk benne. Lehet - meglehetősen torzító - szemüveg, amely megszabja, hogy mit láthatunk, mit foghatunk fel a világból és mit nem. Lehet olyan közvetítő közeg, médium ember és gép között, amely lehetővé teszi ismereteinknek a gép számára feldolgozható formában való kifejezését és a hatékony számítást: például a feladatmegoldás keresési problémává való redukálását. S végül, a tudásreprezentáció lehet töredékes elmélet arról, hogy mit tartunk emberi gondolkodásnak.



Egy reprezentációs nyelvtől elvárható, hogy legyen kifejező és tömör, azaz minden számunkra érdekes dolgot el lehessen benne mondani, legyen független a használat kontextusától, és biztosítsa a hatékony következtetés lehetőségét. Ezeket a követelményeket elégíti ki többé-kevésbé a logika. A logika olyan leíró nyelv, melyben a világ tényeire vonatkozó állításokat mondatok formájában lehet megfogalmazni. A mondatokból mechanikus eljárások alkalmazásával újabb, eddig nem reprezentált mondatok származtathatók. Ez a következtetés. A logika alkalmas arra, hogy kifejezhessük vele ismereteinket, feltételezéseinket, céljainkat, szándékainkat. Alapot ad ahhoz, hogy előzményekből következtetéseket vonjunk le; hogy egyes állítások igazságát vagy hamisságát más állítások igazságértéke fényében ítéljük meg; hogy állítások ellentmondás-mentességét vizsgálhassuk; hogy igazoljuk egy érvelés helytálló voltát.

A mesterséges intelligencia jelenlegi tudásreprezentációs módszereinek többsége az ún. elsőrendű logikán (avagy elsőrendű predikátumkalkuluson) alapul. Ide tartozik az ún. szabály-alapú reprezentáció és következtetés is, amely szakértő rendszerekben messze a leginkább használatos.

## ELŐADÁS

A mesterséges intelligencia alapjai

### Célorientált ágensek

Az eddigiekben a cél keresése

állapottér - környezet

operátor - akció

keresés - célorientált viselkedés

informált keresés - hasznosság orientált viselkedés

Ágens új képességekkel

világról való ismeret - tudásbázis

következtetés képessége - logika

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

## Tudásreprezentáció és Következtetés

### Mi a tudásreprezentáció?

- pótlék (inkább következtetünk, mint tevékenykedünk)
- (erős) szemüveg
- médium (közvetítő közeg)mely
  - lehetővé teszi a hatékony számítást
  - az emberi kifejezést
- (töredékes) elmélet arról, hogy mit nevezünk intelligens gondolkodásnak

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## A tudás leírása

### A leírás szintjei

- tudás szintű (ágens megadása annak révén, amit tud)
- logikai szintű (ahogyan a tudás kódolva van)  
Meghatározza, hogy mit lehet egyáltalán állítani!
- implementáció szintű (pl. programozási nyelv, adatstruktúrák)

### Adott reprezentációs nyelven való leírás

- kifejező erő (Mit lehet egyáltalán állítani?)
- következtetési képesség (Milyen kérdésekre várható válasz?)
- követhetőség (Mennyi időn belül várható válasz?)

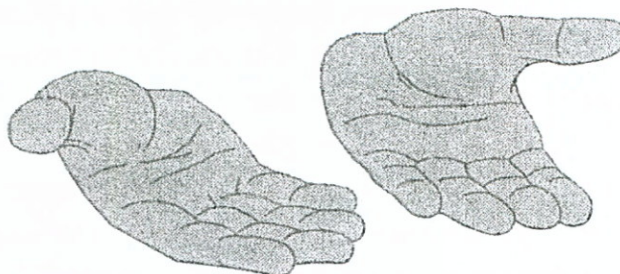
Dr. Szalay Tibor

4. mesterséges intelligencia alapja

**Tudásreprezentáció**



**Modell**



**Alapprobléma**

Miként modellezhető a világ úgy, hogy a modell  
elégleges alapot adjon az intelligens cselekvéshez?

Dr. Szalay Tibor

5. mesterséges intelligencia alapja

### A logika mint reprezentációs nyelv

Mit várunk el egy reprezentációs nyelvtől?

- kifejező és tömör
- egyértelmű
- hatékony következtetést enged meg

### Logika

- mondatok: a világ tényeire vonatkozó állítások megfogalmazása
- következtetés: a mondatokból mechanikus eljárások alkalmazásával újabb, eddig nem reprezentált mondatok származtatása

Dr. Szalay Tibor





## Tények és reprezentációik

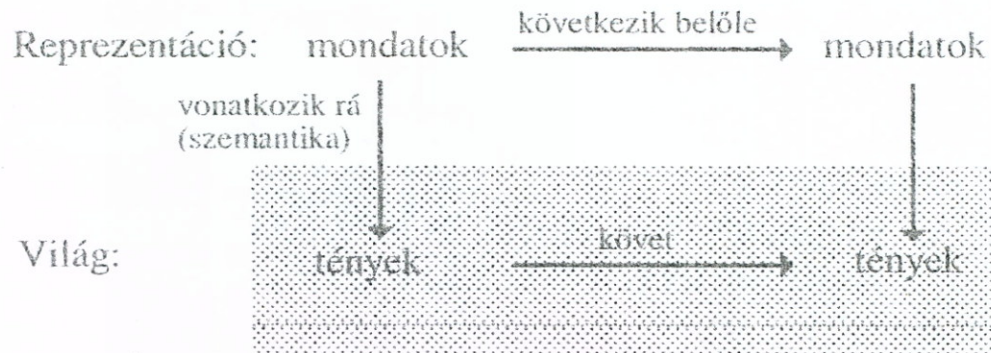
- tény: állítás a világról (igaz vagy hamis)
- reprezentáció
  - kifejezés egy adott nyelven
  - az állításokat kijelentő mondatok fejezik ki
- interpretáció
  - megfeleltetés a tények és a mondatok között
  - általában több interpretáció is lehetséges

## A logika elemei

- szintaxis
  - a nyelv szimbólumai
  - miként lehet mondatokat formálni
- szemantika
  - megszabja, hogy a mondatok a világ mely tényeire vonatkoznak
  - jelentést ad a mondatoknak
- következtetés
  - adott szintaxis és szemantika mellett új mondatokat származtat

A mesterséges intelligencia alapjai

## Következtetés



A következtetéstől elvárjuk, hogy ugyanazon interpretációban az új mondatok is a világ ugyanazon tényeire utaljanak

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Alapvető tulajdonságok és relációk

### Igazság

Egy mondat igaz, ha az, amit leír, valóban előfordul a világban. Egy mondat igazsága függ

- a világ állapotától, és
- az interpretációtól (szemantikától)

### Érvényesség

A mondat igazsága nem függ sem a világ állapotától, sem a szemantikától.

### Kielégíthetlenség/kielégíthetőség

Kielégíthetetlen egy mondat, ha a világ soha nem olyan, mint amilyennek leírja.

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

### Példák

Tautológia (érvényes mondat)

Vagy van erdő az országút mellett vagy nincs

*Ez a mondat érvényes minden interpretációban, bármely világban*

Kielégíthetetlen mondat

Van erdő az országút mellett és nincs erdő az országút mellett

*Nincs olyan interpretáció amelyben ez a mondat kielégíthető*

Egy mondat kielégíthető, ha valamelyik interpretációja valamelyik világban igaz. (Minden érvényes mondat kielégíthető.)

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Következtetési eljárás

Mechanikus szabályokat akarunk alkalmazni, hogy megállapítsuk, mi következik a mondatokból.

premisszák  $\vdash$  konklúzió

TB  $\vdash$  q

Tudásbázis  $\vdash$  mondat

**Jelentése:** a premisszák egy halmazából (tudásbázis) a következtetési eljárás ( $\vdash$ ) egy mondatot (q - konklúzió) állít elő

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Következmény

P1: "esik az eső"

P2: "szabadban vagy"

Q: "vizes leszel"

$S = \{P1, P2\}$

$S \models Q$

Mondatok egy halmaza (S) igaz egy adott interpretációban,  
s ekkor igaz egy újabb mondat is (Q).

Több lehetséges interpretáció: Q igazsága kövesse S igazságát minden lehetséges interpretációban.

(Azaz bármikor, ha S mondatai igazak, akkor igaz Q is.)

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### A következtetés tulajdonságai

biztos: tényleg csak a következményeket állítja elő

Ha  $TB \models q$ , akkor  $TB \models q$

teljes: az összes következményt előállítja

Ha  $TB \models q$ , akkor  $TB \models q$

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

### Következmények gépi kiszámítása

A számítógép nem ismeri a tudásreprezentációs nyelv szemantikáját.

Érvényesség (és kielégíthetetlenség, lsd. később) kulcsszerepben

Ahhoz, hogy eldöntse " $TB \models q$ " igazságát, meg kell mutatnia, hogy a "ha  $TB$  igaz, akkor  $q$ " állítás érvényes; azaz a konklúzió minden interpretációban igaz.

#### Formális következtetés:

érvényes következmények származtatása az interpretációk ismerete nélkül

Dr. Szalay Tibor

## 3.1. Az ítéletkalkulus

A mesterséges intelligencia alapjai

### Az ítéletkalkulus

A legegyszerűbb formális logika

#### Szintaxis

- ítéletváltozók
- ítéletkonstansok (True, False)
- mondatok

Dr. Szalay Tibor

**Mondatok tulajdonságai**

- bármely szimbólum, amely ítéletváltozót vagy konstanst jelöl (pl. P, Q, R, vagy F, T) - atomi mondat
- ha  $\alpha$  egy mondat, akkor mondat még:
  - ( $\alpha$ )
  - $\alpha \wedge \beta$  logikai ÉS, konjunkció
  - $\alpha \vee \beta$  logikai VAGY, diszjunkció
  - $\neg\alpha$  tagadás, negáció
  - $\alpha \Rightarrow \beta$  implikáció ("ha ... akkor ...")
  - $\alpha \Leftrightarrow \beta$  ekvivalencia ("akkor és csak akkor ...")
- mondatok rekurzív módon is konstruálhtók, komplex mondat
  - pl.  $(P \wedge Q) \Rightarrow (\neg R \vee S \vee V)$

**Szemantika**

- interpretáció
  - Egy mondat ítéletváltozóihoz igaz (T) ill. hamis (F) értéket rendel bármely lehetséges módon.
  - Egy mondatnak több interpretációja is lehetséges.
- kiértékelés: a műveleti jelek szemantikája alapján

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
T	T	F	T	T	T	T
T	F	F	F	T	F	F
F	T	T	F	T	T	F
F	F	T	F	F	T	T



A mestersiégés intelligenciára alapjai

**Ekvivalencia, logikai törvények**

Két mondat ekvivalens, ha minden egyes interpretációban ugyanaz az értékük.

Az ekvivalenciát a logikai törvények fejezik ki.

$$1. A \Leftrightarrow B = (A \Rightarrow B) \wedge (B \Rightarrow A)$$

$$2. A \Rightarrow B = \neg A \vee B$$

$$3. A \vee B = B \vee A \text{ valamint } A \wedge B = B \wedge A \text{ (kommutativitás)}$$

$$4. A \vee (B \vee C) = (A \vee B) \vee C \text{ valamint } (A \wedge B) \wedge C \text{ (asszociativitás)}$$

$$5. A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C) \quad (\text{disztributivitás})$$

$$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$$

Dr. Szalay Tibor

A mestersiégés intelligenciára alapjai

$$6. A \vee \text{False} = A$$

$$A \wedge \text{True} = A$$

$$7. A \vee \text{True} = \text{True}$$

$$A \wedge \text{False} = \text{False}$$

$$8. A \vee \neg A = \text{True}$$

$$A \wedge \neg A = \text{False}$$

$$9. \neg(\neg A) = A$$

(A kettős tagadás törvénye)

$$10. \neg(A \vee B) = \neg A \wedge \neg B$$

(De Morgan szabályok)

$$\neg(A \wedge B) = \neg A \vee \neg B$$

$$11. A \vee (A \wedge B) = A$$

$$A \wedge (A \vee B) = A$$

$$12. (A \wedge A) = A$$

$$(A \vee A) = A$$

Az ekvivalenciák könnyen igazolhatók az igazságtábla módszerével.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Következmények ellenőrzése igazságtáblával**

Adott a premisszák egy halmaza (P) és egy konklúzió (K).  
Igaz-e a  $P \Rightarrow K$  mondat minden interpretációban?  
Azaz érvényes-e? — Ha igen akkor K következik P-ből.

Példa: Premisszák:  $P1 \vee P2, \neg P2$  Konklúzió:  $P1$

P1	P2	$P1 \vee P2$	$\neg P2$	$Q = (P1 \vee P2) \wedge \neg P2$	$Q \Rightarrow P1$
T	T	T	F	F	T
T	F	T	T	T	T
F	T	T	F	F	T
F	F	F	T	F	T

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**A módszer előnyei és korlátai**

- mechanikussá tehető a szemantika ismerete nélkül!
- az igazságtáblákon való következtetés teljes
- de exponenciális időt és/vagy helyet igényel ( $2^n$  sor)

Vigyázat! más és más

$\beta \Rightarrow \alpha$  amit az ágens hisz (implikáció)

$TB \models \alpha$  ami helyes (következmény reláció)

$TB \Vdash \alpha$  amit az ágens meg tud tenni (következtetés)

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

## Következtetési eljárások

Az igazságtáblában végrehajtható műveletek bizonyos minták szerint összefoglalhatók.

minták — következtetési szabályok

A következtetési szabályok segítségével átalakíthatjuk a premisszákat alkotó mondatokat.

### 1. $\wedge$ (és) kiküszöbölése

*Egy konjunkcióból következtethető bármely konjunkt.*

$$\frac{a_1 \wedge a_2 \wedge a_3 \wedge \dots \wedge a_n}{a_i}$$

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### 2. kettős tagadás kiküszöbölése

$$\frac{\neg(\neg a)}{a}$$

### 3. $\wedge$ (és) bevezetése

*Egy mondatlistából következtethető ezek konjunkciója.*

$$\frac{a_1, a_2, a_3, \dots, a_n}{a_1 \wedge a_2 \wedge a_3 \wedge \dots \wedge a_n}$$

### 4. $\vee$ (vagy) bevezetése

*Egy mondatból következtethető bármely másikkal levő diszjunkció.*

$$\frac{a_i}{a_1 \vee a_2 \vee a_3 \vee \dots \vee a_n}$$

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### 5. Modus Ponens (Implikáció Kiküszöbölése)

*Egy implikáció és ennek premisszájából következtethető a konklúzió.*

$$\frac{\alpha \Rightarrow \beta, \alpha}{\beta}$$

### 6. Rezolúció ( $\beta$ nem lehet egyszerre igaz és hamis)

*Egy diszjunkcióból, ha az egyik hamis, akkor következtethető, hogy a másik igaz*

$$\frac{\alpha \vee \beta, \neg \beta \vee \gamma}{\alpha \vee \gamma}$$

Bizonyítás: az alkalmazott következtetési szabályok sorozata

Dr. Szalay Tibor

## 3.2. Példák

A mesterséges intelligencia alapjai

### Példák

1. ÉS kiküszöbölése:

$$\frac{\text{esik az eső} \wedge \text{friss a levegő}}{\text{esik az eső}}$$

2. ÉS bevezetése:

$$\frac{\text{esik az eső}, \text{friss a levegő}}{\text{esik az eső} \wedge \text{friss a levegő}}$$

3. VAGY bevezetése

$$\frac{\text{esik az eső}}{\text{esik az eső} \vee \text{friss a levegő}}$$

Dr. Szalay Tibor





A mesterséges intelligencia alapjai

**Példák**

## 4. Modus Ponens

karácsony van  $\Rightarrow$  jön a télapó, karácsony van  
 jön a télapó

## 5. Rezolúció

van ennivaló  $\vee$  étteremben eszünk, nincs ennivaló  $\vee$  itthon vacsorázunk  
 étteremben eszünk  $\vee$  itthon vacsorázunk

Dr. Székely Tíme

A mesterséges intelligencia alapjai

**Példák**

## Igazságtábla

$$\neg A \wedge B \Rightarrow \neg(A \vee \neg B)$$

A	B	$\neg A$	$\neg B$	$\neg A \wedge B$	$A \vee \neg B$	$\neg(A \vee \neg B)$	$A \vee \neg B \Rightarrow \neg(A \vee \neg B)$
T	T	F	F	F	T	F	T
T	F	F	T	F	T	F	T
F	T	T	F	T	F	T	T
F	F	T	T	F	T	F	T

**Érvényes!**

Dr. Székely Tíme

A mesterséges intelligencia alapjai

**Példák**

Bizonyítsa be a következő ekvivalenciát!

$$A \Rightarrow B = \neg A \vee B$$

A	B	$\neg A$	$A \Rightarrow B$	$\neg A \vee B$	ekvivalencia
T	T	F	T	T	T
T	F	T	F	F	T
F	T	T	T	T	T
F	F	T	T	T	T

Dr. Szendrői Tibor

A mesterséges intelligencia alapjai

**Példák**

alszik  $\Rightarrow$   $\neg$ álmos  
 álmos  $\Rightarrow$  rossz-kedvű  
 $\neg$ álmos  $\Rightarrow$   $\neg$ rossz-kedvű  
 alszik

**Modus ponens**

alszik  $\Rightarrow$   $\neg$ álmos, alszik  
 $\neg$ álmos

$\neg$ álmos  $\Rightarrow$   $\neg$ rossz-kedvű,  $\neg$ álmos  
 $\neg$ rossz-kedvű

Dr. Szendrői Tibor



A mesterséges intelligencia alapjai

### Példák - rezolúció

alszik  $\Rightarrow$   $\neg$ álmos  
 álmos  $\Rightarrow$  rossz-kedvű  
 $\neg$ álmos  $\Rightarrow$   $\neg$ rossz-kedvű  
 alszik

$\neg$  alszik  $\vee$   $\neg$ álmos  
 $\neg$  álmos  $\vee$  rossz-kedvű  
 álmos  $\vee$   $\neg$ rossz-kedvű  
 alszik

Tegyük fel: rossz-kedvű

*klóz alak*

$\neg$  alszik  $\vee$   $\neg$ álmos, alszik  
 $\neg$ álmos

álmos  $\vee$   $\neg$ rossz-kedvű,  $\neg$ álmos  
 $\neg$ rossz-kedvű

$\neg$ rossz-kedvű, rossz-kedvű

⊥

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Példák

vezet  $\Rightarrow$   $\neg$ iszik  
 fáradt  $\Rightarrow$   $\neg$ vezet  
 $\neg$ iszik  $\Rightarrow$   $\neg$ elégedett  
 vezet

A)

Mi következik belőle

Modus ponens alkalmazásával?

B)

Következik-e belőle:  $\neg$  fáradt

Rezolúció

Dr. Szalay Tibor

### 3.3. Az elsőrendű logika (predikátum kalkulus)

A mesterséges intelligencia alapjai

#### Elsőrendű logika (predikátumkalkulus)

Alapok: az elsőrendű logika felosztja a világot

- objektumokra, (konstansok, változók, függvények)
- az objektumok tulajdonságaira, (predikátumok)
- az objektumok közti relációkra. (műveletek, kvantorok)

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

#### Szintaxis

1. szimbólum készlet amiből a nyelv felépül

Konstansok	János, A, Kés12
Változók	x, y
Predikátumok	mellete-lévő, apja
Függvények	kor, szomszéd
Logikai műveleti jelek	$\neg$ , $\wedge$ , $\vee$ , $\Rightarrow$ , $\Leftrightarrow$
Kvantorok	$\exists$ , $\forall$

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

2. term-ek melyek objektumokat reprezentálnak  
(konstansok, változók és függvények)

Példa:

$+(2,2)$ ; tulajdonos(Kesz(2)); apa(Miklós)

A függvények term-ekből term-eket építenek

3. mondatok Az objektumokról (azok tulajdonságairól és relációiról) való állításokat fejezik ki. Igazságértékük van.

Példa:

egyenlő( $+(2,2)$ , 4); foglalkozása(Miklós, kertész)

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Mondatok tulajdonságai**

predikátumok (Állítások objektumokról)

$PI. = (+(2,2), 4)$ ;  $apja(Miklós, Péter)$

összetett mondatok (Mondatok logikai műveletekkel összekötve).

Ha mondat P és Q, akkor az  $\neg P$ ,  $P \wedge Q$ ,  $P \vee Q$ ,  $P \Rightarrow Q$ ,  $P \Leftrightarrow Q$ .

kvantifikáció

Ha P egy mondat és x egy változója, akkor mondat

- $\forall xP$  (univerzális kvantifikáció),
- $\exists xP$  (egzisztenciális kvantifikáció)

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Szemantika

(hogyan adunk jelentést a mondatoknak?)

Interpretáció — Megfeleltetés a formális nyelv (reprezentáció) és a világ elemei között.

<u>Világ</u>	<u>Reprezentáció</u>
Miklós kertész.	foglalkozása(Miklós,kertész)
Miklósnak van egy zöld sapkája.	hirtokol(Miklós,sapka15) $\wedge$ zöld(sapka15)
Miklós látta Sárát.	látta(Miklós,Sára)
Miklós látta Sára apját.	látta(Miklós,apa(Sára))
	$\exists x$ apja(x,Sára) $\wedge$ látta(Miklós,x)
Minden keréksz szereteti a napot.	$\forall x$ keréksz(x) $\Rightarrow$ szereti(x,Nap)

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Mondatok kiértékelése

- predikátumok

Állítások, melyek vagy igazak vagy hamisak.

- összetett mondatok

Kiértékelés a tagmondatok igazságértéke alapján, az ítéletkalkulus igazságtáblája szerint.

- kvantifikáció

$\forall xP$  akkor és csak akkor igaz,  
ha  $Px$  minden lehetséges értékére igaz.

Pl.  $\forall x$  delfin(x)  $\Rightarrow$  emlős(x)

$\exists xP$  akkor és csak akkor igaz,  
ha  $x$ -nek van olyan értéke, amire  $P$  igaz.

Pl.  $\exists x$  emlős(x)  $\Rightarrow$  vízben-él(x)

Dr. Szalay Tibor



### 3.3.1. Következtetési eljárások

A mesterséges intelligencia alapjai

#### Következtetés az elsőrendű logikában

Következtetési eljárás

Következtetési szabályok segítségével adott mondatokból újabb (ún. jól formált) mondatokat hoz létre.

tétel: a származtatott mondat

bizonyítás: a következtetési szabályok sorozata

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

#### Dedukció

Biztos és teljes következtetés

- biztos (egy következtetési szabály)
 
$$\text{Ha } TB \vdash q, \text{ akkor } TB \models q$$
- teljes (a következtetési szabályok egy halmaza), ha az összes következményt elő tudják állítani
 
$$\text{Ha } TB \models q, \text{ akkor } TB \vdash q$$

*emlékeztető*

ellenpélda: abdukció (nem biztos! Csak ekvivalencia esetén)

$$p \Rightarrow_k q$$

$$\frac{}{q \Rightarrow_d p}$$

esik  $\Rightarrow_k$  vizes-a-fű (kauzális)

vizes-a-fű  $\Rightarrow_d$  esik (diagnosztikai)

Dr. Szalay Tibor

### A természetes levezetés

#### Következtetési szabályok

Sok szabály lehetséges; a mondatokból egyes elemeket kiküszöbölnek, ill. bevezetnek.

implikáció kiküszöbölése (*Modus Ponens*)

implikáció bevezetése

Feltesszük, hogy P igaz; ha ennek segítségével ki tudjuk mutatni, hogy Q is igaz, akkor  $P \Rightarrow Q$  minden feltétel nélkül igaz (de nem Q!).

#### Példa

esik  $\Rightarrow$  vizes-a-fű

vizes-a-fű  $\Rightarrow$  nem-kell-locsolni

Tegyük fel, hogy esik. Ekkor

(1) esik, esik  $\Rightarrow$  vizes-a-fű  
vizes-a-fű

(2) vizes-a-fű, vizes-a-fű  $\Rightarrow$  nem-kell-locsolni  
nem-kell-locsolni

(3) nem-kell-locsolni,  
esik  $\Rightarrow$  nem-kell-locsolni

Most az esik törlendő, míg az esik  $\Rightarrow$  nem-kell-locsolni hozzáadható a tudás-bázishoz.



A mesterséges intelligencia alapjai

**Reductio ad absurdum (alkalmazás)**

Feltesszük, hogy  $\neg P$  igaz; ha ennek segítségével ellentmondást tudunk kimutatni, akkor  $P$  igaz. (Igazolás cáfolat révén.)

Példa

esik  $\Rightarrow$  vizes-a-fű

$\neg$ vizes-a-fű (ezt úgy írjuk, hogy vizes-a-fű  $\Rightarrow \perp$ )

Tegyük fel, hogy esik. Ekkor

(1) esik, esik  $\Rightarrow$  vizes-a-fű  
vizes-a-fű

(2) vizes-a-fű, vizes-a-fű  $\Rightarrow \perp$   
 $\perp$

(3) Így állítható, hogy  $\neg$ esik.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Tulajdonságok**

- biztos és teljes (ítéletkalkulus, elsőrendű logika)
- "természetes" az emberi gondolkodás számára
- nagy elágazási tényező
  - több szabály igen sokféle módon alkalmazható
  - automatikus következtetés (tételbizonyítás) számára nem alkalmas

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Következtetés kvantorokkal

Miként helyettesíthetők a változók term-ekkel, melyek egyedi objektumokra utalnak?

#### Helyettesítés (szubsztitúció)

jelölése:  $\text{subst}(\Theta, \alpha)$  — ahol  $\Theta$  helyettesítést alkalmazunk egy  $\alpha$  mondatra, és  $\{v_1/t_1, \dots, v_n/t_n\}$  helyettesítési lista megadja, hogy az egyes objektumokat milyen termekkel helyettesítettük.

#### Példa

$\text{subst}(\{x/\text{Miklós}, y/\text{hal}\}, \text{megeszi}(x, y)) = \text{megeszi}(\text{Miklós}, \text{hal})$

A mesterséges intelligencia alapjai

### Helyettesítési szabályok

(Együtt alkalmazhatók a korábbi szabályokkal)

Cél: kvantorok kiküszöbölése ill. bevezetése.

#### $\forall$ kiküszöbölése

$\forall y$  megeszi(Miklós, y)  
megeszi(Miklós, hal25)...  
megeszi(Miklós, kacsa2)

#### $\exists$ bevezetése

megeszi(Miklós, hal25)  
 $\exists x$  megeszi(Miklós, x)

#### $\exists$ kiküszöbölése

$\exists x$  megeszi(Miklós, x)  
megeszi(Miklós, ennivaló)

Nebet ad egy olyan objektumnak, ami kielégíti a mondatban megfogalmazott állítást (ún. Skolem konstans).

17. szabály 10. oldal



A mesterséges intelligencia alapjai

### Miként automatizálható a következtetés folyamata?

Következtetés mint keresés

kiindulási állapot = axiómák konjunkciója

operátorok = következtetési szabályok

cél állapot = bizonyítandó tétel

Probléma: túl nagy elágazási tényező

- sok és sokféle módon alkalmazható következtetési szabály
- ismert tények kombinálása: exponenciális növekedés
- kiút: más, jobb keresési tér egyetlen, általánosabb következtetési szabállyal

Dr. Szalay Tibor

### 3.3.2. Általánosított Modus Ponens

A mesterséges intelligencia alapjai

#### Általánosított Modus Ponens

$$\frac{p_1', p_2', \dots, p_n', (p_1 \wedge p_2 \wedge \dots \wedge p_n \Rightarrow q)}{\text{subst}(\Theta, q)}$$

–  $p_i, p_i', q$  egyszerű (atomi) mondatok, és

–  $\text{subst}(\Theta, p_i') = \text{subst}(\Theta, p_i)$

Példa:

premisszák:

$\forall x \text{ macska}(x) \wedge \text{kóbor}(x) \Rightarrow \text{bolhás}(x)$

macska (Mirci)

kóbor (Mirci)

helyettesítés:

$\Theta = \{x/\text{Mirci}\}$

konklúzió:

$\text{subst}(\{x/\text{Mirci}\}, \text{bolhás}(x)) = \text{bolhás}(\text{Mirci})$

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Hogyan használható az általánosított *Modus Ponens*?

A következtetés iránya

### 1. előre-láncolás

- újabb és újabb állításokat ad a tudásbázishoz
- adatvezérelt
- baj: nem irányul egy adott probléma megoldására, sok irreleváns konklúziót állíthat elő

### 2. Hátra-láncolás (*alaplóműködés*)

- benne van a tény a tudásbázisban?
- ha nincsen, akkor olyan konklúziókat keres, amelyek a tényre illeszkednek, és megpróbálja igazolni azok premisszáit.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Példa - előre láncolás

- (1) A macskák szeretik a halat.  
 $\forall x \text{ macska}(x) \Rightarrow \text{szereti}(x, \text{hal})$
- (2) A macskák mindent megesznek, amit szeretnek.  
 $\forall x \forall y \text{ macska}(x) \wedge \text{szereti}(x, y) \Rightarrow \text{megeszi}(x, y)$
- (3) Mirci egy macska.  
 $\text{macska}(\text{Mirci})$

Igaz-e az alábbi állítás?

- (4) Mirci megeszi a halat.  
 $\text{megeszi}(\text{Mirci}, \text{hal})$

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

**Megoldás - modus ponens (előre-láncolás)**

A) - 1,3:

$$\frac{\forall x \text{ macska}(x) \Rightarrow \text{szereti}(x, \text{hal}), \text{macska}(\text{Mirci})}{\text{szereti}(\text{Mirci}, \text{hal})}$$

B) - 2,3:

$$\frac{\forall x \forall y \text{ macska}(x) \wedge \text{szereti}(x, y) \Rightarrow \text{megeszi}(x, y), \text{macska}(\text{Mirci})}{\forall y \text{ szereti}(\text{Mirci}, y) \Rightarrow \text{megeszi}(\text{Mirci}, y)}$$

C) - A,B

$$\frac{\forall y \text{ szereti}(\text{Mirci}, y) \Rightarrow \text{megeszi}(\text{Mirci}, y), \text{szereti}(\text{Mirci}, \text{hal})}{\text{megeszi}(\text{Mirci}, \text{hal})}$$

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Megoldás - modus ponens (hátra-láncolás)**A)  $\text{megeszi}(\text{Mirci}, \text{hal})$  — nincs a tudásbázisbanB)  $\text{megeszi}(\text{Mirci}, \text{hal})$  $\text{Macska}(\text{Mirci}), \text{szereti}(\text{Mirci}, \text{hal})$  $\text{Macska}(\text{Mirci}), \text{szereti}(x, \text{hal}), \forall x \text{ macska}(x)$ 

Melyiket érdemes használni?

az elágazási tényező dönti el, néha jobb az előre-láncolás

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Az általánosított *Modus Ponens* problémái**

- a keresés irányítása
- nem teljes

**Ezzel szemben: *Rezolúció***

teljes  
(többé-kevésbé) jól automatizálható  
egyetlen szabály



Dr. Szalay Tibor

**3.3.3. Általánosított *Rezolúció***

A mesterséges intelligencia alapjai

**Az Általánosított *Rezolúció***

szabálya

$$\frac{P_1 \vee P_2 \vee \dots \vee P_n \quad \neg P_1 \vee Q_2 \vee \dots \vee Q_m}{P_2 \vee \dots \vee P_n \vee Q_2 \vee \dots \vee Q_m} \quad (\text{rezolvens})$$

*Modus Ponens* mint egyszerű *rezolúció*

$$\frac{\alpha \Rightarrow \beta, \alpha}{\beta}$$

$$\frac{\neg \alpha \vee \beta, \alpha}{\beta}$$

Dr. Szalay Tibor



**Bizonyítási eljárás: a tagadás cáfolatával****Kielégíthetlenség kimutatása**

Ha adott az ellentmondásmentes axiómák egy halmaza TB,  
ahhoz, hogy igazoljuk,  $TB \models W$ ,  
mutassuk meg, hogy  $(TB \cup \{\neg W\})$  kielégíthetetlen.

Ha a mondatok halmaza kielégíthetetlen, a rezolúció  
ellentmondásra jut.

**Alkalmazás**

- cél tagadása, az axiómákhoz való hozzáadása,
- ellentmondás kimutatása
- kiindulás: normál alak (klóz)

Dr. Szalay Tibor

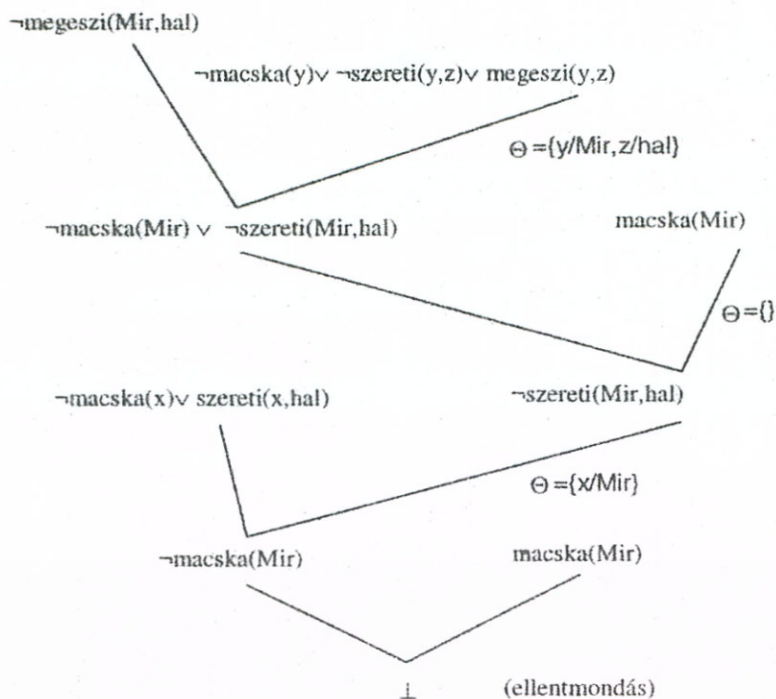
**3.3.4. Példák****Példa - rezolúció**

- (1) A macskák szeretik a halat.  
 $\forall x \text{ macska}(x) \Rightarrow \text{szereti}(x, \text{hal})$
  - (2) A macskák mindent megesznek, amit szeretnek.  
 $\forall x \forall y \text{ macska}(x) \wedge \text{szereti}(x, y) \Rightarrow \text{megeszi}(x, y)$
  - (3) Mirci egy macska.  
 $\text{macska}(\text{Mirci})$
- Igaz-e az alábbi állítás?
- (4) Mirci megeszi a halat.  
 $\text{megeszi}(\text{Mirci}, \text{hal})$

Dr. Szalay Tibor



A mesterséges intelligencia alapjai



**Bizonyítás**

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Példa 2

Egy színész elfoglaltsága

- (1) előadás(kedd)  $\vee$  előadás(péntek)
- (2) vívóedzés(kedd)
- (3) vívóedzés(péntek)
- (4) előadás(kedd)  $\wedge$  vívóedzés(kedd)  $\Rightarrow$  ütközés
- (5) előadás(péntek)  $\wedge$  vívóedzés(péntek)  $\Rightarrow$  ütközés

Következi-e mindebből ütközés?

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

Tömörebb leírásban, konjunktív normál alakban:

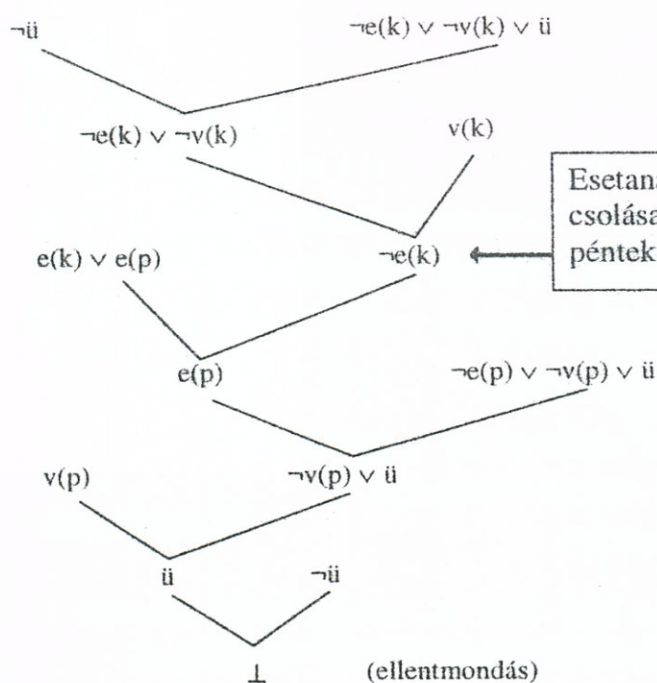
- (1)  $e(k) \vee e(p)$
- (2)  $v(k)$
- (3)  $v(p)$
- (4)  $\neg e(k) \vee \neg v(k) \vee \ddot{u}$        $(e(k) \wedge v(k) \Rightarrow \ddot{u})$
- (5)  $\neg e(p) \vee \neg v(p) \vee \ddot{u}$        $(e(p) \wedge v(p) \Rightarrow \ddot{u})$

Cél:  $\ddot{u}$  (az ütközés) kimutatása.

Bizonyítása:  $\neg \ddot{u}$  ellentmondásra vezet.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai



**Bizonyítás**

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Rezolúciós stratégiák

### Rezolúció

- 1) Tagadd a célállítást és add azt a tudásbázishoz.
- 2) Alakítsd át a mondatokat klóz formára.
- 3) Amíg az üres klózt elő nem állítod, alkalmazd ismételten a rezolúciós szabályt.

### Nemdeterminisztikus eljárás

melyik két klóz-ra alkalmazzuk a szabályt?

több egyesítő helyettesítés esetén melyiket válasszuk?

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Szűkítő feltétel a rezolválandó párok kiválasztására

1. szélességben-először keresés
  - első szintű klóz: axiómák és a cél tagadása
  - az összes N-edik szintű klózt előállítja, mielőtt az (N+1) szintűeket
2. támogató halmaz stratégiája
  - legalább az egyik szülő a cél tagadása, vagy annak egyik leszármazottja (célirányos bizonyítás)
3. lineáris rezolúció
  - legalább az egyik szülő az alaphalmazból, vagy a másik szülő elődei közül

Dr. Szalay Tibor



## 4. input rezolúció

- legalább az egyik szülő az alaphalmazból

## 5. egységklóz rezolúció

- legalább az egyik szülő egységklóz  
(redukálja a rezolvens méretét)

**Egyszerűsítési stratégiák**

tautológiák kiküszöbölése

$$PI. P(f(A)) \vee \neg P(f(A))$$

befoglaló klózok elhagyása

$$\text{subst}(\Theta, Q) \supset \text{subst}(\Theta, P)$$

Dr. Szalay Tibor

**Példa**

- (1) Bárki, aki tud olvasni, az tud írni.

$$\forall x [O(x) \Rightarrow L(x)] \quad \neg O(x) \vee L(x)$$

- (2) A delfinek nem tudnak írni.

$$\forall x [D(x) \Rightarrow \neg L(x)] \quad \neg D(y) \vee \neg L(y)$$

- (3) Van olyan delfin, amelyik intelligens.

$$\exists x [D(x) \wedge I(x)] \quad D(A) \quad (3a)$$

$$I(A) \quad (3b)$$

**Bizonyítandó:**

- (4) Van aki intelligens, de nem tud olvasni.

$$\exists x [I(x) \wedge \neg O(x)] \text{ tagadva: } \neg I(z) \vee O(z)$$

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Levezetés:

- |    |          |                                   |
|----|----------|-----------------------------------|
| 5) | (3b)+(4) | $\Theta = \{z/A\}: O(A)$          |
| 6) | (1)+(5)  | $\Theta = \{x/A\}: L(A)$          |
| 7) | (2)+(6)  | $\Theta = \{y/A\}: \neg D(A)$     |
| 8) | (3a)+(7) | $\perp$ (üres klóz, ellentmondás) |

Dr. Csabó Tibor



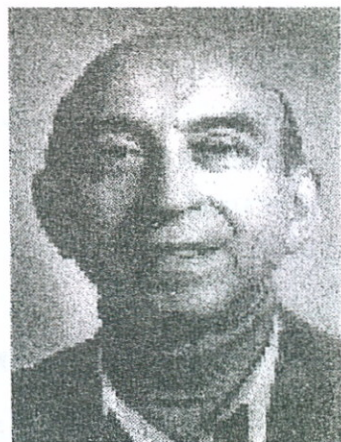
## 4. A BIZONYTALAN TUDÁS KEZELÉSE

Mindennapos tapasztalataink szerint a logika csak egyféle tükre az emberi gondolkodásnak, észjárásaink lehetnek egészen mások, mint ami a logika segítségével formalizálható. Egyrészt képesek vagyunk bizonytalan, hiányos, akár ellentmondásos ismeretekre hagyatkozva is racionálisan dönteni és cselekedni, másrészt gondolkodásunk él az emlékezet adta lehetőségekkel. Ez lehetővé teszi, hogy hasznosítani tudjuk a korábbi feladatok megoldásait, hogy el tudjuk kerülni korábbi hibáinkat, s hogy képesek legyünk analógiák és hasonlatok segítségével is a feladatmegoldásra.

Reális körülmények között nem várhatjuk el, hogy a világ minden kérdésünkre kategorikus választ adjon; nincs mindig mód arra, hogy biztosan megtudjuk, hogy egy tény valóban igaz-e vagy hamis. A világról való tudásunk bizonytalan, s ezt a bizonytalanságot valamilyen módon ki kell fejeznie a világról szóló reprezentációnak, s tekintetbe kell vennie a reprezentáció felett működő következtetésnek is. Ezt teszik lehetővé a valószínűségi logikák és az ún. *fuzzy* logika.

### 4.1. Fuzzy logika

Az elmúlt tíz évben jelentek meg a fuzzy rendszerek a tudományos élet különböző területein, valamint a mérnöki és gazdasági alkalmazásokban és több helyen kezdik kiszorítani a hagyományos technikákat. A fuzzy logika felhasználási területe a háztartási és szórakoztatóelektronikai eszközöktől (mosógépek, videokamerák, légkondicionálók, porszívók), a forgalomirányító berendezéseken keresztül (hajók navigációs rendszere, metrók irányítása) egészen az ipari automatizálásig terjed. (lásd. I. táblázat) 1965-ben jelent meg Lotfi Zadeh-nek, a fuzzy logika szülőatyjának a "Fuzzy sets" című műve. Elmélete kezdetben kevés támogatóra talált, mivel ebben az időszakban sikerült a hardver fejlesztőknek első processzoraikat előállítani melyek Zadeh kifejezésével élve "sikeresen tudtak különbséget tenni 0 és 1 között", és ennek következtében a hagyományos matematikai logika került előtérbe. Hamarosan kiderült, hogy világunk nem csak egyszerűen nullákkal és egyesekkel leírható igaz-hamis állításokból áll és felmerült az igény egy



Lotfi Zadeh



flexibilis logikára mely képes ábrázolni a valós világ összefüggéseit és ezzel a fuzzy logika előtérbe kerülhetett.

A fuzzy logika módot ad a modell alapú rendszertervezés intuícióval és mérnöki heurisztikákkal való támogatására. A mindennapi nyelv elemeit használja a rendszer kívánt viselkedésének leírására. Bonyolult rendszerek tervezéséhez és optimalizálásához nyújt hatékony eszközt a szokásos matematikai modellek használata nélkül. Folytonos változókkal képes dolgozni és a tradicionális kétértékű logikával szemben kezelni tud parciális és többértékű igazságokat is. Használata különösen akkor előnyös, ha az aktuális probléma matematikailag nehezen reprezentálható az adatok hiányos volta, vagy a folyamat túlságos összetettsége miatt. Robosztus rendszerek felépítését teszi lehetővé, melyek hiányos és zajos bementi adatok alapján is képesek hibamentes működést biztosítani.

Felhasználási területek	Alkalmazások száma
Folyamatszabályozás	49
Háztartási berendezések	8
Méréstechnika/szenzorok	29
Gépkocsi elektronika	46
Robottechnika	19
Képfeldolgozás	31
Diagnosztika	16
Egyéb	121
Összesen	319

1. táblázat: A fuzzy rendszerek felhasználási területei Japánban 1987–1990

A valós világ nyelvének használata lehetővé teszi, hogy az emberi logika becslő természetével vértessük fel a számítógépet. További előnyei mutatkoznak meg, ha már meglévő szimbolikus tudást akarunk beépíteni a rendszerbe. A nyelvi modellezés használata nagyban megnöveli a rendszer módosíthatóságát, ez pedig rövid fejlesztési ciklushoz és könnyű programozhatósághoz vezet. Zadeh szerint: "Majdnem minden esetben elérhetjük ugyanazt az eredményt fuzzy logika nélkül is, de fuzzy logikával ez gyorsabb, egyszerűbb és olcsóbb".

#### 4.1.1. Fuzzy halmazok

Az általános halmazelmélet értelmezése szerint a halmaz meghatározott tulajdonságú objektumok összessége. Halmazokat elemeik felsorolásával vagy a definiáló feltételek körülírásával lehet megadni. A definíció az objektumo-



kat két csoportba osztja, a halmazhoz tartozó és a halmazhoz nem tartozó elemekre, vagyis egy jól definiált, merev és éles határt húz meg a két csoport között. A fuzzy halmazelmélet egyfajta bizonytalanság bevezetésével ezen merev határok elmosására törekszik. Egyes értelmezések szerint a fuzzy halmazok a hagyományos halmazelmélet kiterjesztéseként, ill. általánosításaként is felfoghatók, más vélemények szerint néhány fuzzy teória teljesen sajátosnak és egyedinek tekinthető. A fuzzy halmazok interfészként szolgálnak az emberek által használt nyelvi változók és a számítógépeknek megfelelő mennyiségi megközelítés között.

Ha  $X$  az  $x$  objektumok egy csoportja, akkor az  $X$ -en értelmezett  $A$  fuzzy halmaz a következő rendezett párok halmaza:

$$A = \{(x, \mu_A(x)) : x \in X\}. \quad (1) \quad !$$

A  $\mu_A(x)$  neve *tagsági függvény*, értéke pedig az  $x$  tagsági aránya az  $A$ -ban. A  $\mu_A(x)$  leképezi az  $X$ -et az  $M$  tagsági térbe ( $\mu_A: X \rightarrow [0,1]$ ). Ha  $M$  csak 0 és az 1 pontokból áll, akkor az  $A$  nem fuzzy. A tagsági függvény értékei véges valós számok a  $[0,1]$  intervallumból. Ha egy jellemzőt tagsági függvények segítségével írunk le, akkor a tagsági függvény értéke azt a fokot jelenti, hogy egy változó bizonyos értéke mennyire felel meg a jellemzőnek.

#### 4.1.2. Fuzzy relációk

A fuzzy relációk elméletének alapjait szintén az általános halmazelméletben megtalálható relációk képezik. A hagyományos értelemben vett relációk két vagy több halmaz elemei között lévő kapcsolat, összefüggés, viszony, kommunikáció jelenlétét vagy annak hiányát jellemzik. A fuzzy relációk hasonlóan értelmezhetők azzal a kiegészítéssel, hogy a viszony erősségének, ill. milyenségének foka a fuzzy halmazokhoz hasonlóan tagsági függvényekkel jellemezhető. Tehát nemcsak az összefüggés meglétéről vagy hiányáról adnak információt, hanem annak mértékéről is, így a kapcsolatot sokkal árnyaltabban jellemzik.

#### 4.1.3. Fuzzy következtető rendszer

alkalmazhatjuk:

Korábban említettük már, hogy a fuzzy relációk és az ezzel kapcsolatos műveletek komoly jelentőséggel bírnak a gyakorlati élet szempontjából. A fuzzy rendszerek a szakértő rendszerekhez hasonlóan szintén szabályalapúak, ahol a tudás-reprezentáció alapjait a következtetési



szabályok jelentik, de más elgondolást használnak a szabályok nyelvi elemeinek reprezentálására. Az alapvető céljuk is hasonló, vagyis a szakértők tudását HA ... AKKOR típusú szabályok formájában megjeleníteni a rendszerben. Ha például egy  $x$  bemeneti és egy  $y$  kimeneti változónk van, akkor a következő, logikai implikáción alapú szabályt

Ha az input  $x$  "értéke"  $A$ , akkor az output  $y$  "értéke" legyen  $B$ , ahol  $A$  és  $B$  fuzzy részhalmazok az  $X, Y$  halmazon  $\mu_A$  és  $\mu_B$  tagsági függvényvel.

A nyelvi kifejezésekkel adott szabályok kiértékelését fuzzy következtetésnek nevezzük. A legtöbb fuzzy logikán alapuló rendszer produkciós szabályokat használ a nyelvi változók és az események között kapcsolat leírására. A szabályok feltétel része (HA rész) egy vagy több ÉS vagy VAGY nyelvi kapcsolattal összekapcsolt feltételből áll.

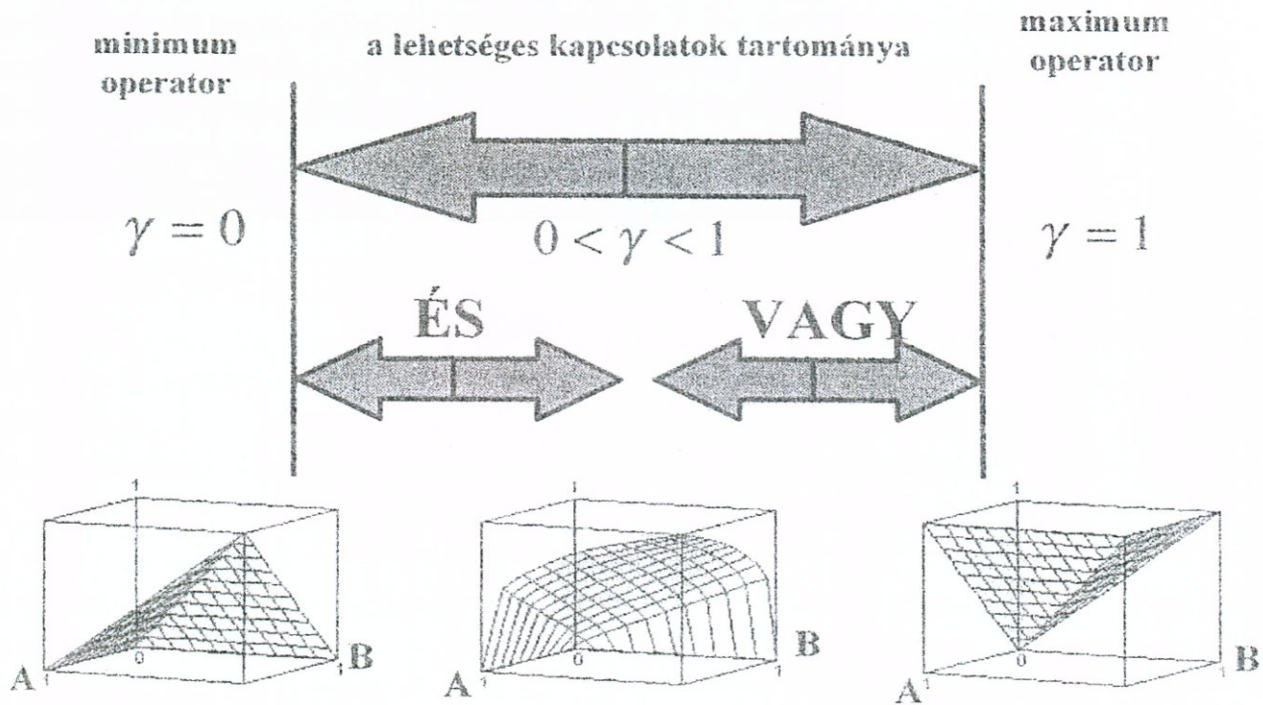
A fuzzy következtetés folyamata két alapvető részre osztható: *aggregációra* és *kompozícióra*. Az aggregáció egyszerű fuzzy operátorokat és relációkat felhasználva meghatározza, hogy milyen mértékben teljesül az adott szabály feltétel oldala. A kompozíció a feltételek érvényességéből adja meg a következmények érvényességét.

Egyszerű alkalmazásokban kielégítő a szabályok feltétel oldalán az ÉS kapcsolatnál a minimum, míg VAGY kapcsolatnál a maximum használata, az ide vonatkozó fuzzy műveleteknek megfelelően (lásd az 1. Gyakorlatot). Egyes esetekben viszont a lehetőségeink korlátozásához vezet a csupán ÉS és a VAGY kapcsolatokkal felépített szabályrendszer. Igény van egy árnyaltabb összefüggést kifejező, ún. kompenzációs operátor bevezetésére. Ilyen például a GAMMA kompenzációs operátor. A GAMMA operátor matematikai modellje a következő:

$$\mu = \left( \prod_{i=1}^m \mu_i \right)^{1-\gamma} \left( 1 - \prod_{i=1}^m (1 - \mu_i) \right)^{\gamma} \quad (2)$$

A végletet jelentő minimum, ill. maximum operátor között sajátos módon "torzítja" a döntési felületet a gamma tényező függvényében, melynek értéke a  $[0,1]$  intervallumban változhat.





3. ábra: A GAMMA kompenzációs operátor működése

#### 4.1.4. Fuzzy Logic Controller

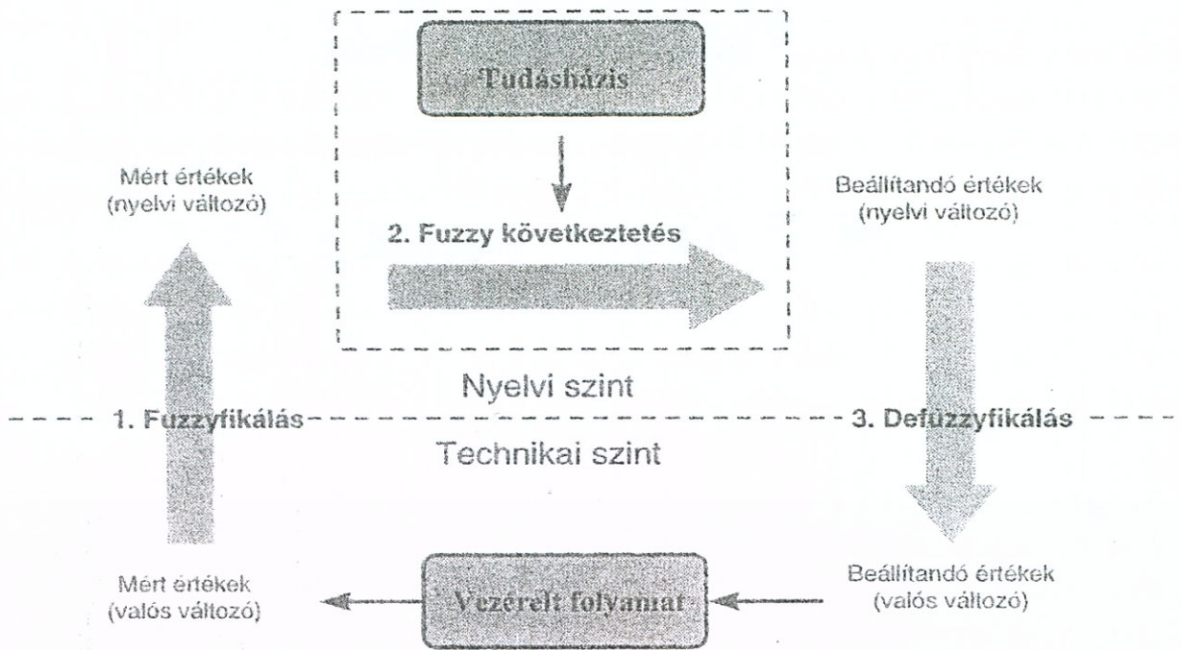
Ahogy a fuzzy halmazok az általános halmazelmélet kiterjesztésének tekinthetők és a fuzzy logika a klasszikus kétértékű vagy matematikai logika kiterjesztése, továbbá ahogy a kétértékű logika és a halmazelmélet között párhuzamok vonhatók, ugyanígy megtalálható ez a fuzzy logika és a fuzzy halmazok között is. Például összefüggést találhatunk az unió halmazművelet és a VAGY logikai operátor, vagy a metszet művelet és az ÉS operátor között.

Szabályozási alkalmazásokban a fuzzy logikát a vezérlési stratégia köznapin nyelven való leírására használjuk. Minden szabályozási stratégia célja a kívánt kimenetek eléréséhez megfelelő bemenetek meghatározása, de mivel a szabályozni kívánt rendszer elemei nem képesek értelmezni a fuzzy nyelvi elemeket, kétirányú fordításra van szükség a pontos érték és a nyelvi változók között.

A fuzzy logika vezérlésekben történő alkalmazáskor a folyamatban lejátszódó alapvető eseményeket a 4. ábra szemlélteti. Három alapvető lépést különböztethetünk meg.

- **Fuzzyfikálás:** valós érték lefordítása nyelvi kifejezésekre, melyeket fuzzy halmazok reprezentálnak.
- **Fuzzy következtetés:** nyelvi változók közötti összefüggéseket definiáló szabályhalmaz kiértékelése

- **Defuzzyfikálás:** fuzzy következtetés eredményét fordítja vissza nyelvi kifejezésekről valós értékekre.



4. ábra A fuzzy logic controller alapvető struktúrája

#### 4.1.5. Fuzzyfikálás

A fuzzy logikát a valós világ nyelvének használata teszi igazán hatékony eszközzé. A *nyelvi változók* alkalmazása kulcsfontosságú a fuzzy szakértői rendszerek és fuzzy kontroll rendszerek területén. A nyelvi változók értékei szavak, mondatok, vagyis hétköznapi nyelvi elemek lehetnek. A legtöbb valós helyzetben egy precíz válasz nem feltétlenül jelent optimális megoldást a problémára. A fuzzy logika kiterjeszti a szigorú igaz/hamis osztályozást olyan értékekkel, mint például a *nagyjából igaz* és a *teljesen rossz*. Ezeket a kifejezéseket nyelvi változóknak, magát a folyamatot fuzzyfikálásnak hívjuk.

#### 4.1.6. Következtető rendszer és a szabályalapú tudásbázis

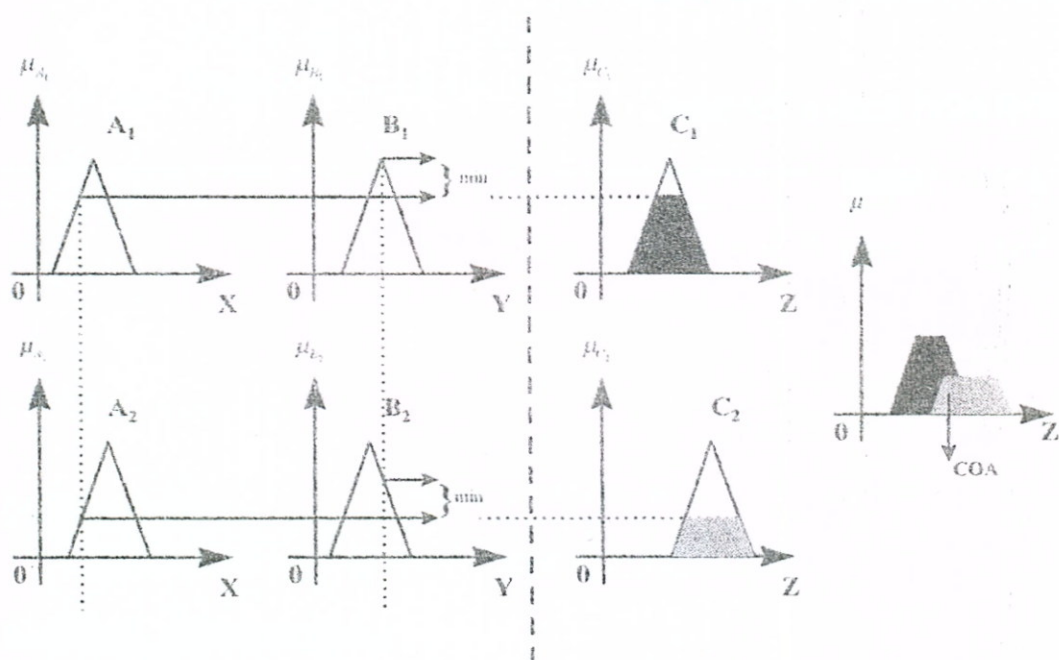
A fuzzy szabályok alkalmazásának számos előnye van a hagyományos értelemben vett szabályalapú szakértői rendszerekkel szemben. A klasszikus szakértői rendszerekben található szabályok hátránya, hogy érvényességüket pontosan kell definiálni, mivel rövid úton ellentmondásba keveredhetünk, márpedig ilyen pontos definíciók az emberi gondolkodás intuitív volta miatt



nehezen előállíthatóak. A fuzzy rendszerek esetében a nyelvi változókkal reprezentált mennyiségek lehetővé teszik a részleges igazságot egy vagy akár több halmazban is. A következtető modul a szabályrendszer kiértékelését végzi, mely folyamat elméleti alapjait az előző fejezetben láthattuk.

#### 4.1.7. Defuzzifikáció

A fuzzy szabályok kiértékelésével kapott eredmény szintén fuzzy, ezen értékek valós értékké való visszafordítását nevezzük defuzzifikálásnak. Különböző eljárások állnak rendelkezésünkre. Mivel egyidejűleg több kifejezés is érvényes lehet, ezért a defuzzifikálási eljárás feladata, hogy ezek együttes figyelembevételével hozza meg a döntését. A három legáltalánosabban elterjedt eljárás a Center-of-Maximum (COM), a Center-of-Area (COA) és a Mean-of-Maximum (MOM). A COM módszer esetén a tagsági függvények maximumhelyének súlyozott átlagaként számolja ki a kimeneti értéket, ahol a súlyok a megfelelő szabályok kimeneti értékei.



5. ábra A defuzzifikálás folyamata Center of Area eljárás esetén

A COA módszer a talán a leggyakrabban használt eljárásnak tekinthető, numerikus integrálást használ a tagsági függvények által határolt terület kiszámítására. Az így kiszámított területek közös súlypontját határozza meg, amely a defuzzifikálás végeredménye. A MOM módszer a legnagyobb érvényességi fokkal rendelkező kifejezés maximumaként számolja ki a kimenetet.

Az 5. ábra a COA defuzzifikálás folyamatát mutatja be, ha a rendszer tudásbázisában az alábbi két szabály szerepel:



HA  $x = A_1$  ÉS  $y = B_1$  AKKOR  $z = C_1$  ;  
 HA  $x = A_2$  ÉS  $y = B_2$  AKKOR  $z = C_2$

## ELŐADÁS

A megerőséges intelligencia alapjai

### Bizonytalan tudás kezelése

#### Bizonytalan tudás forrása

a környezet

tudatlanság ill. lustaság

#### Módszerek

Valószínűségi logika

Bizonyossági tényezők alkalmazása

Fuzzy logika

Dr. Székely István



## 4.2. A valószínűségi logika

A mesterséges intelligencia alapjai

### Valószínűségi logika (alapok)

#### valószínű állítások

$P(p)=x$  Ha semmi más információ nem áll rendelkezésre,

a p mondat valószínűsége x.

#### valószínűségi eloszlás

X véletlen változó milyen  $\langle x_1, x_2, \dots, x_n \rangle$  értékeket milyen valószínűséggel vehet fel. Pl.

$$P(\text{idő}=\text{napos}) = 0.6$$

$$P(\text{idő}=\text{szeles}) = 0.3$$

$$P(\text{idő}=\text{esik}) = 0.08$$

$$P(\text{idő}=\text{havazik}) = 0.02$$

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

#### következtetés

Hogyan kell egy új mondatot felcímkézni?

#### feltételes valószínűség

$P(p|q)=y$  Ha egyedül azt tudjuk, hogy q igaz, akkor p valószínűsége y.

Vigyázat!  $P(p|q)$  általában  $\neq P(q \Rightarrow p)$

Dr. Szalay Tibor

### Valószínűségi axiómák

A racionális döntés szabályai

- 1)  $P(p \wedge q) = P(p) \cdot P(q|p)$
- 2)  $P(\neg p) = 1 - P(p)$
- 3) Ha  $p \equiv q$ , akkor  $P(p) = P(q)$

### Együttes valószínűségi eloszlás

- kimerítő módon megadja az összes állítás valószínűség eloszlását
- elemi esemény:  $X_i$  változó felvesz egy bizonyos értéket
- $P(X_1, X_2, \dots, X_n)$  külön-külön  $X_i$ -k minden lehetséges értékeire
- Leírható egy  $n$ -dimenziós táblázatban.



A mesterséges intelligencia alapjai

**Példa**

	csikorgás	$\neg$ csikorgás
törés	0.04	0.06
$\neg$ törés	0.01	0.89



$$P(\text{törés}) = 0.04 + 0.06 = 0.1$$

$$P(\text{törés} \vee \text{csikorgás}) = 0.04 + 0.06 + 0.01 = 0.11$$

Ismeretlen A állítás valószínűsége, ha ismerjük B-ét:

$$\begin{aligned} P(\text{törés} | \text{csikorgás}) &= P(\text{törés} \wedge \text{csikorgás}) / P(\text{csikorgás}) \\ &= 0.04 / (0.04 + 0.01) = 0.8 \end{aligned}$$

Dr. Szalay Tibor

**4.3. A bizonyossági tényezők módszere**

A mesterséges intelligencia alapjai

**Bizonyossági tényezők**

A bizonytalanság kezelésének régebbi módszere.

Állításokhoz rendelünk egy bizonyossági tényezőt  
(*certainty factor*):  $cf = a \in [-1, 1]$

-1: biztosan hamis

1: biztosan igaz

0: nem tudunk semmit

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Következtetés

A bizonyossági tényezőket kombinálni kell:

$$cf(B_1 \wedge B_2 \wedge \dots \wedge B_n) = \min[cf(B_i)]$$

$$cf(B; B \Rightarrow A) = cf(B) * cf(B \Rightarrow A) \text{ ha } cf(B) > 0 \\ = 0 \text{ egyébként}$$

ha egy állítás több úton is levezethető, akkor értéke:

$$cf(x,y) = x + y - xy, \quad \text{ha } x > 0 \text{ és } y > 0; \\ = x + y + xy, \quad \text{ha } x < 0 \text{ és } y < 0; \\ = (x + y) / (1 - \min(|x|, |y|)), \quad \text{ha } x * y < 0.$$

Dr. Szalay Tibor

## 4.4. Fuzzy algebra alapjai

A mesterséges intelligencia alapjai

"Minden hagyományos logika feltételezi, hogy pontos szimbólumokkal dolgozik. Ezért nem is alkalmazhatók erre a földi életre, csak egy elképzelt mennyei létre."

Bertrand Russel

Dr. Szalay Tibor



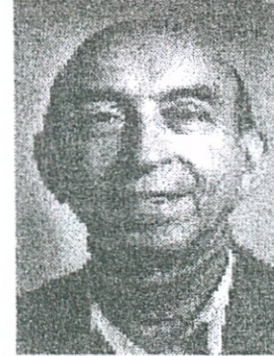
A mesterséges intelligencia alapjai

## Fuzzy Logika

FUZZY (angol) = HOMÁLYOS,  
ZAVAROS,  
KÓCOS  
BIZONYTALAN

LOFTI ZADEH (1965) - Fuzzy sets

"... a hagyományos matematikai logika sikerrel tesz különbséget a 0 és az 1 között ..."



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## DEFINÍCIÓ

**Fuzzy halmaz:** Ha  $X$  az  $x$  objektumok egy csoportja, akkor az  $X$ -en értelmezett fuzzy halmaz a következő rendezett párok halmaza:

$$A = \{(x, \mu(x)) : x \in X\}$$

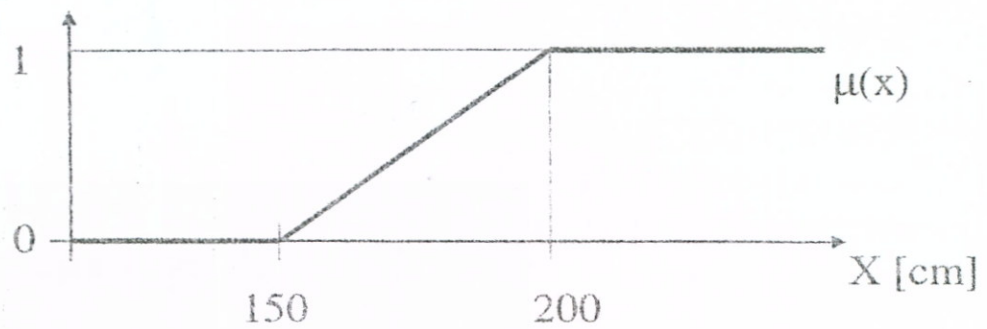
**Tagsági függvény** ( $\mu(x)$ ): leképezi az  $X$ -et az  $M$  tagsági értékbe, megmutatja az  $x$  tagsági arányát az  $A$ -ban.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Példa:** Az eléggé magas emberek (X) fuzzy halmazának tagsági függvénye:

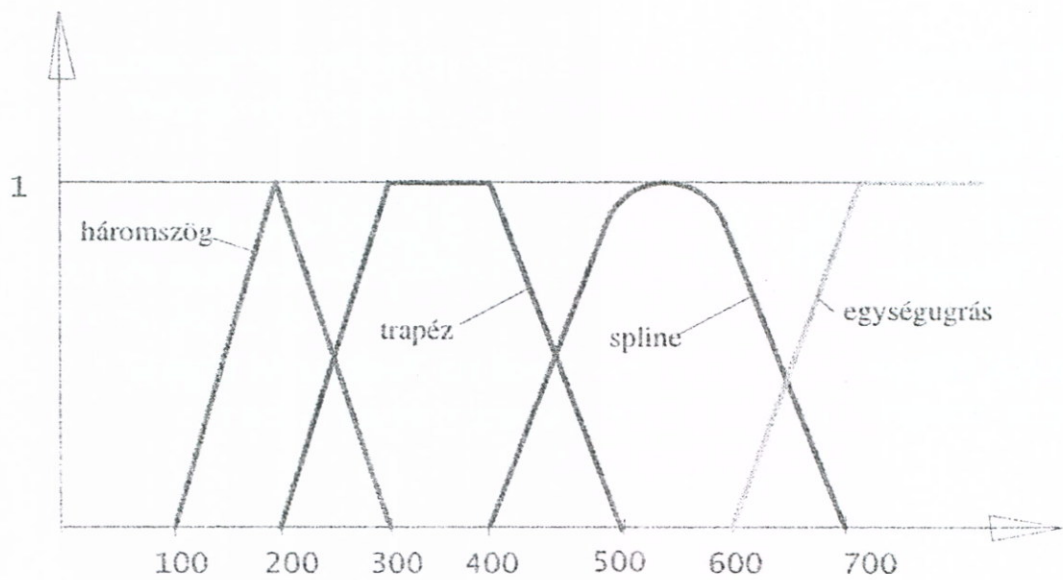
$$\mu(x) = \begin{cases} 0, & \text{ha } x \leq 150 \\ 50x - 3, & \text{ha } 150 < x \leq 200 \\ 1, & \text{ha } 200 < x \end{cases}$$



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Néhány jellemző tagsági függvényalak**



Dr. Szalay Tibor



A mesterséges intelligencia alapjai

## Igazságérték

### egyszerű állítás esetén

Nem *True* vagy *False*, hanem egy szám 0 és 1 között.

$$T(\text{True}) = 1 \text{ és } T(\text{False}) = 0$$

$$T(\text{magas-ember}(\text{Miklós})) = 0.8$$

### összetett mondat

Igazságérték számítása a komponensek értékeiből, a fuzzy logika szabályai szerint:

$$T(A \wedge B) = \min(T(A), T(B))$$

$$T(A \vee B) = \max(T(A), T(B))$$

$$T(\neg A) = 1 - T(A)$$

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Következtetés

Ha az  $A$  fuzzy mennyiség olyan, hogy abból következik a  $B$  fuzzy mennyiség, akkor létezik (nem egyetlen) olyan  $R$  relációs mátrix, amely "tudja" ezt a következtetést, vagyis  $B = A \circ R$ . Ilyen tulajdonságú a következő mátrix:

$$\mu_R(x, y) = \min \{ \mu_A(x), \mu_B(y) \},$$

A mátrix neve *következtető mátrix*, jele  $R = A \times B$ .

Dr. Szalay Tibor

### Következtetések egyesítése

Legyenek  $R_1 = A_1 \times B_1$  és  $R_2 = A_2 \times B_2$  következtetési mátrixok. Az az  $R$  reláció, amelynek tagsági függvénye

$$\mu_R(x,y) = \max \{ \mu_{R_1}(x,y), \mu_{R_2}(x,y) \},$$

egyszerre tudja közelítőleg az  $A_1 \Rightarrow B_1$  és az  $A_2 \Rightarrow B_2$  következtetéseket.

### A fuzzy következtetés előnyei - 1

→ A szabályozás leírása könnyű. Csupán szabályok rendszerét kell kidolgozni egy-egy folyamatra a szabályozási téren belül.

→ A fuzzy szabályozás komplex bemenetet, kimenetet tud kezelni.

→ A fuzzy következtetési rendszeren belül a műveletek párhuzamosan hajthatók végre.



A mesterséges intelligencia alapjai

## A fuzzy következtetés előnyei - 2

→A fuzzy következtetés az egyszerű és logikus nyelvi szabályok-ból áll (ha  $A = B$ , akkor  $C = D$ ), azaz a mindennapi nyelvet lehet használni.

→Több feltétel is szerepelhet a HA részben.  
(pl. ha  $X1=A$ ,  $X2=B$ ,  $X3=C$ , akkor  $Y=D$ )

→A szabályok alakja független a feltételektől.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

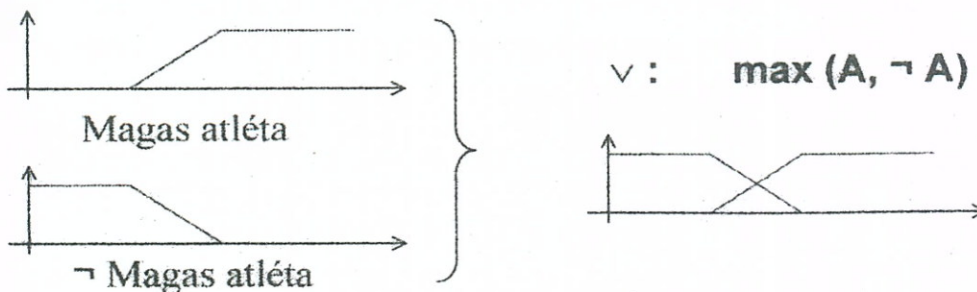
## Probléma

nem konzisztens az ítéletkalkulussal és az elsőrendű logikával

Ahol elvárható, hogy  $A \vee \neg A \Leftrightarrow True$

Ezzel szemben itt  $T(A \vee \neg A) \neq T(True)$

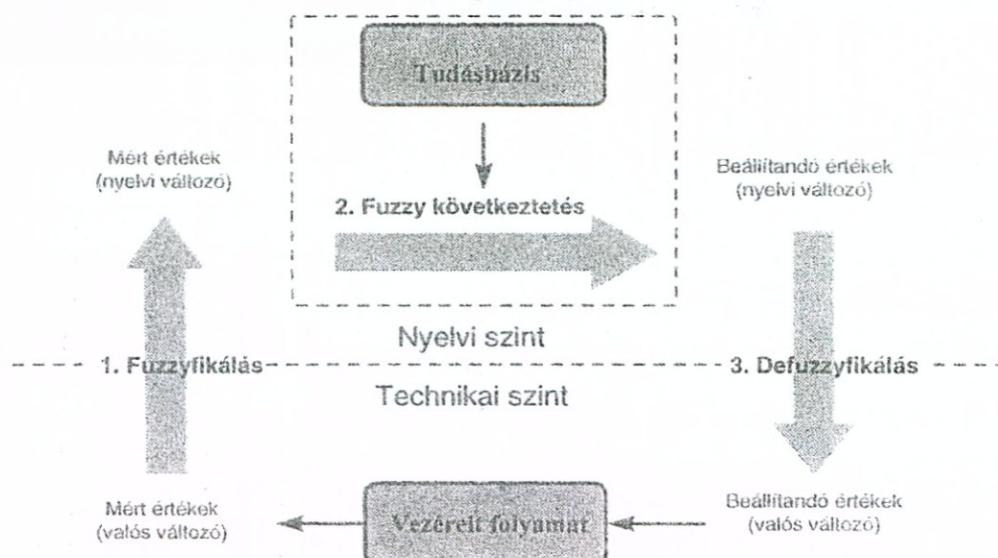
**Példa:** Magas atléta  $\vee$   $\neg$  Magas atléta



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Alkalmazások: szabályozási feladatok



- kis szabálybázis, hangolható paraméterek (tanulás)
- nincs következtetési lánc

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### A fuzzy szabályozás elemei

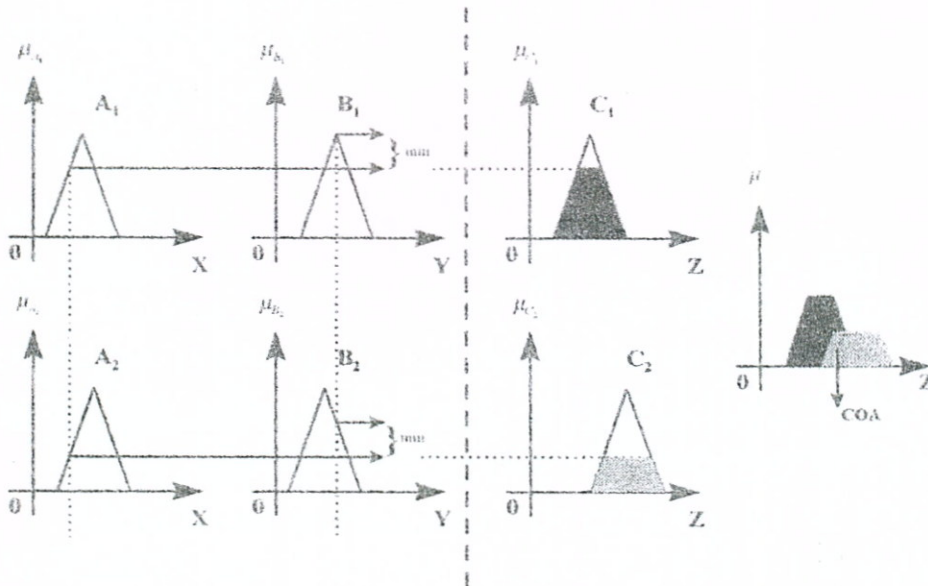
- Fuzzyfikálás:** valós érték lefordítása nyelvi kifejezésekre, melyeket fuzzy halmazok reprezentálnak.
- Fuzzy következtetés:** nyelvi változók közötti összefüggéseket definiáló szabályhalmaz kiértékelése
- Defuzzyfikálás:** fuzzy következtetés eredményét fordítja vissza nyelvi kifejezésekről valós értékekre.

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

### Defuzzifikáció



Dr. Szalay Tibor

### 4.5. Példák

A fuzzy elméletben két alapvető következtetési szabály van, a *kompozíció* és a *következtetés*.

#### Reláció

Az  $R$   $X \times Y$ -on értelmezett fuzzy mennyiség egy a két jelenség kapcsolatát kifejező *reláció*. A tagsági értékeket tartalmazó mátrix neve *relációs mátrix*.

#### Példa:

Mi a kapcsolat egy nyelvtanfolyamon szerzett osztályzat és az olvasási képesség között? Egy adott idegen nyelvű szöveget kell a résztvevőknek előolvasni, az olvasási időt mérve létrehozhatjuk a relációs mátrixot.



	elégtelen	elégséges	közepes	jó	jeles
1. sor : 0.5-1 perc	0	0	0.1	0.4	0.9
2. sor : 2-5 perc	0	0.2	0.6	0.4	0.3
3. sor : 5-nél több perc	0	1	0.4	0.2	0

## Kompozíció

Legyen  $A$  egy  $R$ -en értelmezett fuzzy mennyiség,  $R$  pedig egy  $X \times Y$ -on értelmezett fuzzy reláció. A  $B$  fuzzy mennyiséget az  $A$  következményének nevezzük, ha  $Y$ -on értelmezett és

$$\mu_B(y) = \max \min \{ \mu_R(x,y), \mu_A(x) \} \quad \forall y \in Y \quad (9)$$

A művelet neve *kompozíció*, jele  $B = A \circ R$ .

### Példa :

Az  $R$  reláció legyen az előző példában megadott, az  $A$  pedig az, hogy a résztvevő "gyorsan olvas",  $(0.8, 0.3, 0)$  tagsági értékekkel. A kompozíció műveletet alkalmazva megállapíthatjuk a kapott osztályzat tagsági függvényét. Ez  $(0, 0.2, 0.3, 0.4, 0.8)$ , tehát a jeles osztályzatnak a legnagyobb az esélye, az elégtelen pedig kizárható.

## Következtetés

Ha az  $A$  fuzzy mennyiség olyan, hogy abból következik a  $B$  fuzzy mennyiség, akkor létezik (nem egyetlen) olyan  $R$  relációs mátrix, amely "tudja" ezt a következtetést, vagyis  $B \equiv A \circ R$ . Ilyen tulajdonságú a következő mátrix:

$$\mu_R(x,y) = \min \{ \mu_A(x), \mu_B(y) \} \quad \forall x \in X, \forall y \in Y \quad (10)$$

A mátrix neve *következtető mátrix*, jele  $R = A \times B$ .

### Példa:

Tegyük fel, hogy nem áll rendelkezésünkre az előző példában szereplő mátrix, de tudjuk, hogy

"HA a résztvevő NAGYON LASSAN olvas,  
AKKOR ő a tanfolyamon ROSSZ jegyet kapott"

A "ROSSZ jegy" tagsági függvénye  $(1, 1, 0.4, 0, 0)$ , a "NAGYON LASSAN olvas" tagsági függvénye  $(0, 0, 1)$ , a következtető mátrix :

0	0	0	0	0
0	0	0	0	0
1	1	0.4	0	0



Ez a reláció sajnos nem tartalmaz elegendő információt olyan kérdések eldöntésére, mint hogy mi van a KEVÉSSÉ LASSAN olvasó résztvevőkkel. Éppen ezért szükséges egy, az ilyen különböző kérdéseket eldöntő, több következtetés egyesítésére.

### Következtetések egyesítése

Legyenek  $R_1 = A_1 \times B_1$  és  $R_2 = A_2 \times B_2$  következtetési mátrixok. Az az  $R$  reláció, amelynek tagsági függvénye

$$\mu_R(x,y) = \max \{ \mu_{R_1}(x,y), \mu_{R_2}(x,y) \} \quad \forall x \in X, \quad \forall y \in Y$$

egyszerre tudja közelítőleg az  $A_1 \Rightarrow B_1$  és az  $A_2 \Rightarrow B_2$  következtetéseket.

Az előző példában levő szabályhoz tegyük hozzá azt, hogy

"HA a résztvevő NORMÁL SEBESSÉGGEL olvas  
AKKOR az osztályzata valószínűleg NEM ROSSZ"

A "NEM ROSSZ jegy" tagsági függvénye (0,0.1,0.6,0.2,0), a "NORMÁL SEBESSÉGGEL olvas" tagsági függvénye (0,0.8,0.2), a következtető mátrix

0	0	0	0	0
0	0.1	0.6	0.2	0
0	0.1	0.2	0.2	0

A két következtetést egyesítve már sokkal sokoldalúbb következtetési mátrixot kapunk:

0	0	0	0	0
0	0.1	0.6	0.2	0
1	1	0.4	0.2	0

Az egyesített következtetési mátrix már nem adja vissza egyik kiinduló következtetést sem pontosan, az eredményt ugyanis most az összes felvett ismeret alapján kapjuk. És éppen ez az, amit egy következtető rendszertől elvárunk.

Következtető mátrixot létrehozhatunk több feltételes szabályok esetén is. Például az  $(A \text{ és } B) \Rightarrow C$  szabályt  $A \Rightarrow (B \Rightarrow C)$  alakban írva először létrehozzuk a  $B \Rightarrow C$  szabály mátrixát, majd újra az unió (max)

műveletet használva kapunk egy háromdimenziós mátrixot. Ezt használhatjuk hagyományos kétfeltételes esetek kiértékelésére, azaz először a  $B$ , majd az  $A$  mennyiséggel "szorozva". (A sorrend nem lényeges, csak a mátrix megfelelő oldalait nem szabad összecsereelni.) Ha a feltételek egyike hosszabb ideig állandó marad, például valamilyen körülményt rögzítünk egy kísérletsorozatban, ennek tagsági függvényével megszorozva a mátrixot a kapott kisebb dimenziójú mátrix megőrzi következtető tulajdonságát, az egyfeltételes következtetéshez használhatjuk.

A mesterséges intelligencia alapjai

### 1. háztartási gépek

(vízmelegítő, légkondicionáló, mosógép)

**PÉLDA**  $\Rightarrow$  "AISAI" márkájú mosógép (japán)

A ruha súlyának megmérése után egy adott mennyiségű vizet és mosószert adagolt. Ezzel mosva optikai szenzor figyelte a víz zavarosodását, ebből következtetett a szennyeződés fokára és fajtájára, majd fuzzy következtetési szabályok alapján választotta ki a legjobb mosóprogramot.

Dr. Györfi Tibor



A mesterséges intelligencia alapjai

## 2. kamerák (videokamera, fényképezőgép)

PELDA ⇒ Electronic Image Stabilizer

Az elektromos képstabilizáló működését fuzzy szabályok biztosítják, közülük kettő:

- Ha a mozgásvektorok különböző irányúak és különbségük növekszik, akkor egy mozgó tárgy van a képen.
- Ha a mozgásvektorok egyformák, és különbségük kicsi, akkor a mozgást a kéz remegése okozta.

Dr. Szabó Tibor

A mesterséges intelligencia alapjai

## 3. Folyamat felügyelet, folyamat szabályozás

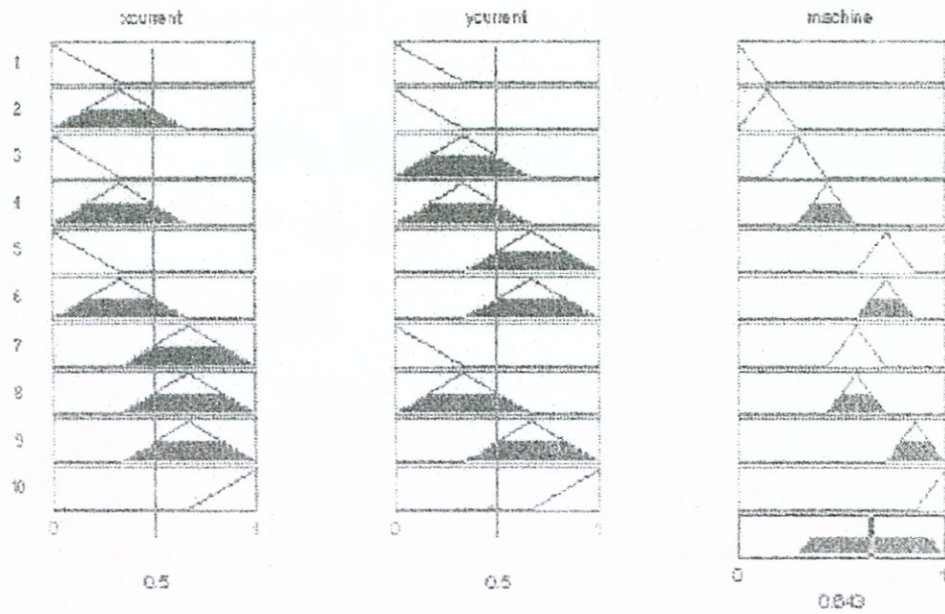
PELDA ⇒ Marási folyamat felügyelete



A fuzzy döntési rendszer felépítése

Dr. Szabó Tibor

A mesterséges intelligencia alapjai



A szabályok és következmények szemléltetése

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

A rendszer 8 különböző állapotot ismer fel

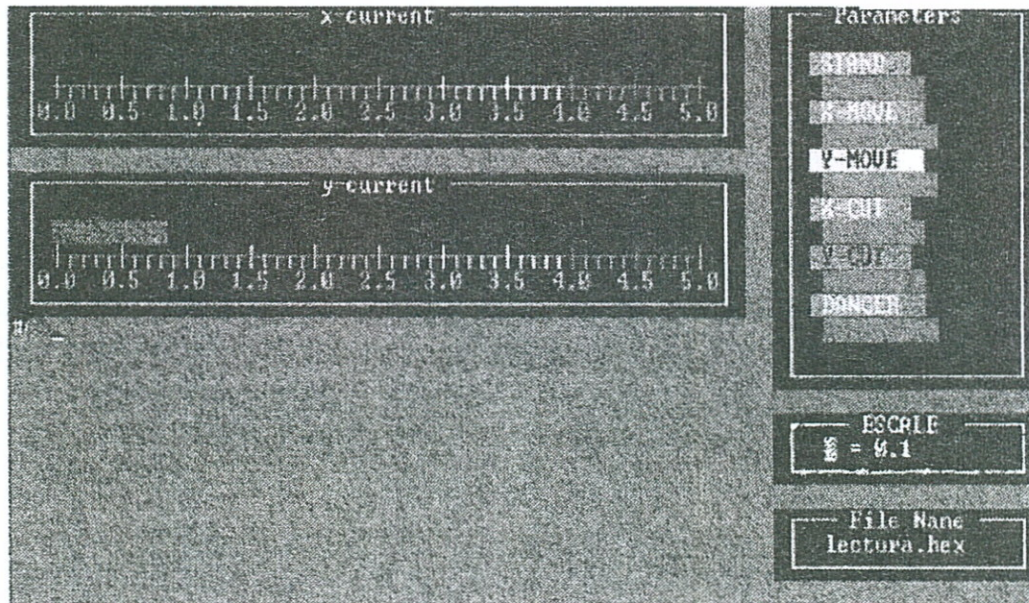
- Áll
- Hossz irányú mozgás
- Kereszt irányú mozgás
- Mozgás mindkét irányba
- Hossz irányú megmunkálás
- Kereszt irányú megmunkálás
- Megmunkálás mindkét irányba
- Veszély

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

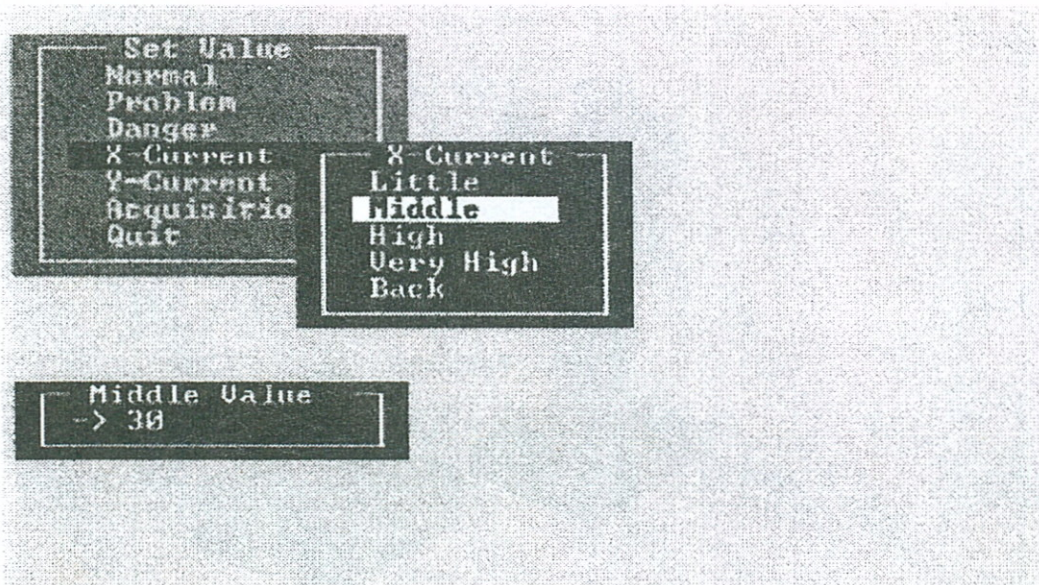
## A rendszer kommunikációs megjelenése



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## A rendszer paramétereinek beállítása



Dr. Szalay Tibor



A mesterséges intelligencia alapjai

#### 4. Robotirányítás, hibrid vezérlés

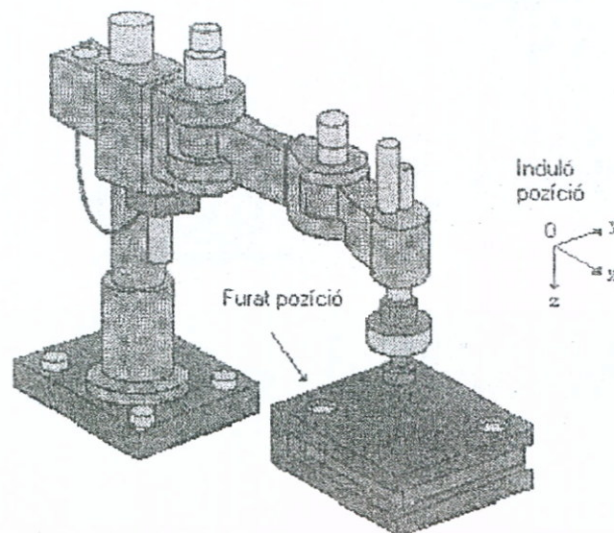
PELDA → Csap - furat illesztési feladat

Feltétel: A keresés a furat meghatározottan szűk környezetében kezdődik, előre meghatározott irányban. A bevezető kúp meredeksége változó.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

Leírás: A robot mozgását a megfogóba integrált 3 komponenses erőmérő szenzor jelei alapján egy fuzzy szabályozó rendszer irányítja.



Dr. Szalay Tibor



A mesterséges intelligencia alapjai

**Feladat:** Fuzzy szabályok megfogalmazása a pontos helyzet meghatározásához.

**Bemenő változók**

- három erőkomponens (FX, FY, FZ)  
7 fuzzy szint (N-, K-, A-, O, A+, K+, N+)

**Kimeneti változók**

- három irányban lépésköz (MX, MY, MZ)  
(vagy három csulóban az inkremensek)  
7 fuzzy szint (N-, K-, A-, O, A+, K+, N+)

Dr. Szalay Tibor

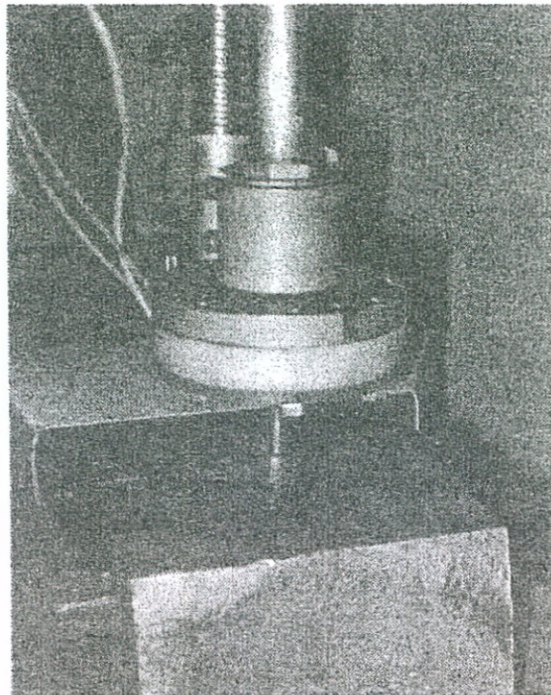
A mesterséges intelligencia alapjai

**Példák mozgásvezérlő fuzzy szabályokra**

1. Ha FX az O és FY az O és FZ is O,  
akkor MZ az N+.
2. Ha FX az O és FY az K- és FZ az K-,  
akkor MY az K- és MZ az K+.
3. Ha FX az N+ és FY az K+ és FZ az A-,  
akkor MX az A+ és MY az K+ és MZ az N+.

Dr. Szalay Tibor

A mesterséges intelligencia alapja



Dr. Szalay Péter

**Megvalósítás**

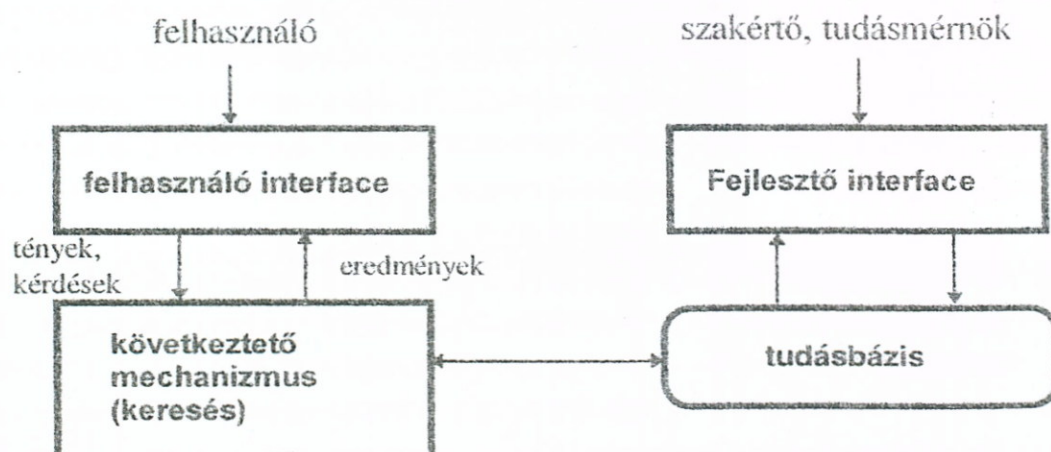


## 5. A TUDÁSREPREZENTÁCIÓ ALKALMAZOTT MÓDSZEREI

A mesterséges intelligencia eddigi legsikeresebb alkalmazásai az ún. szakértő rendszerek. A szakértő rendszer (*expert system*) olyan számítógépprogram, melynek működését egy adott szakterülettel kapcsolatos ismeretek határozzák meg. A szakértő rendszerben a reprezentált ismeretek és az azokkal dolgozó következtetési mechanizmus szét van választva. Hasonló dolgot fejez ki a tudás-alapú rendszer megjelölés is, azzal a különbséggel, hogy itt a reprezentált ismeretek köre szélesebb; a hangsúly az ismeretek deklaratív ábrázolásán van. A szakértő rendszerek jellegzetessége, hogy általában egy szűk, erősen behatárolt szakterület részletes ismereteinek felhasználásával működnek. Sikerük titka éppen ebben rejlik: általában a szakértő tudás minden nehézség ellenére jobban behatárolható, megfogalmazható és formalizálható, mint a köznapi ismeretek.

Ma már több ezer szakértő rendszer működik (optimista becslések szerint számuk meghaladja a tízezret). A leginkább kedvelt területek a gyógyászat, az üzleti élet és a gyártás, ahol a szakértő rendszerek elsősorban diagnosztikai, tanácsadási, konfigurációs, ütemezési és irányítási feladatokat látnak el. Mindezek mellett ma már aligha akad olyan terület, ahol nem került volna alkalmazásra valamilyen szakértő rendszer.

A szakértő rendszerek fő komponensei (1) a tudásbázis, (2) a következtető mechanizmus, és (3) a felhasználói interface. A rendszerek szerves része még a telepítéskor leválasztható fejlesztő környezet (lásd 1. ábra).



1. ábra: Szakértő rendszerek felépítése



Alább a szakértő rendszerekben leginkább használatos tudásábrázolási és következtetési módszereket tekintjük röviden át. Az áttekintés kiterjed magára a tudásreprezentáció problémájára, a hagyományos szimbolikus módszerekre, valamint az ideghálós reprezentáció eszközeire.

### 5.1. A tudásábrázolás és következtetés módszerei

A tudásábrázolás (vagy -reprezentáció), és a hozzá elválaszthatatlanul kapcsolódó következtetés a mesterséges intelligencia egyik alapkérdése: miként modellezhető a világ úgy, hogy a modell elégséges alapot adjon a racionális cselekvéshez? Maga a tudásreprezentáció lehet a világ valódi dolgait helyettesítő pótlék, amely lehetővé teszi, hogy következtetéseket vonjunk le a világról, s ne csak tevékenykedjünk benne. Lehet – meglehetősen torzító – szemüveg, amely megszabja, hogy mit láthatunk, mit foghatunk fel a világból és mit nem. Lehet olyan közvetítő közeg, médium ember és gép között, amely lehetővé teszi ismereteinknek a gép számára feldolgozható formában való kifejezését és a hatékony számítást: például a feladatmegoldás keresési problémává való redukálását. S végül, a tudásreprezentáció lehet töredékes elmélet arról, hogy mit tartunk emberi gondolkodásnak.

Egy reprezentációs nyelvtől elvárható, hogy legyen kifejező és tömör, azaz minden számunkra érdekes dolgot el lehessen benne mondani, legyen független a használat kontextusától, és biztosítsa a hatékony következtetés lehetőségét. Ezeket a követelményeket elégíti ki többé-kevésbé a logika. A logika olyan leíró nyelv, melyben a világ tényeire vonatkozó állításokat mondatok formájában lehet megfogalmazni. A mondatokból mechanikus eljárások alkalmazásával újabb, eddig nem reprezentált mondatok származtathatók. Ez a következtetés. A logika alkalmas arra, hogy kifejezhessük vele ismereteinket, feltételezéseinket, céljainkat, szándékainkat. Alapot ad ahhoz, hogy előzményekből következtetéseket vonjunk le; hogy egyes állítások igazságát vagy hamisságát más állítások igazságértéke fényében ítéljük meg; hogy állítások ellentmondás-mentességét vizsgálhassuk; hogy igazoljuk egy érvelés helytálló voltát.

A mesterséges intelligencia jelenlegi tudásreprezentációs módszereinek többsége az ún. elsőrendű logikán (avagy elsőrendű predikátumkalkuluson) alapul. Ide tartozik az ún. szabály-alapú reprezentáció és következtetés is, amely szakértő rendszerekben messze a leginkább használatos.



## 5.2. 1.1.2. Szabály-alapú reprezentáció és következtetés

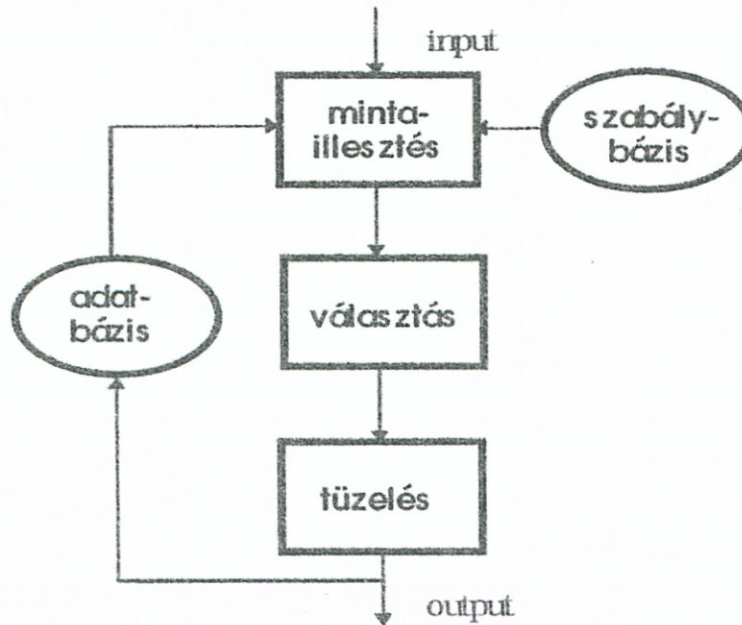
Szabály-alapú (avagy produkciós) rendszerek tudásbázisa két részből áll: az adatbázisból, és a szabálybázisból. Az adatbázis – melyet szokás munkatárként (*working memory*) is emlegetni – elemi tényállítások halmaza. Ez rögzíti mindazon tényeket, melyeket egy probléma megoldásának adott pillanatában a világról és magának a feladatnak a megoldásáról ismerünk. A zárt világ feltételezése érvényben van: azaz minden állítás, ami nincs benne az adatbázisban, hamisnak tekintendő.



A tudásbázis másik fő része a szabálybázis, mely ha – akkor típusú szabályokból áll. Egy szabály egy feltétel – akció páros, aminek olvasata: "bár-mikor, ha látod, hogy a feltétel teljesül, akkor hajtsd végre az akciót!". A feltétel oldal kifejezi azokat a körülményeket, melyek között egy szabály egyáltalán aktivizálódhat. A feltételt egy minta (*pattern*) írja le, amely adatbázisbeli tényekre vonatkozik. A mintában tények logikai kapcsolata definiálható (ÉS, VAGY), illetve tagadás fogalmazható meg. A minta tartalmazhat változókat is. Az akció oldal adja meg, hogy mi történjék a szabály aktivizálódása (ún. tüzelése) nyomán. Egy szabálynak kétféle hatása lehet: módosíthatja az adatbázist újabb elemi tényállítások hozzáadásával vagy korábbiak törlésével, illetve, amolyan mellékhatásként, input/output feladatokat láthat el.

Szabály-alapú rendszerekben az alapvető következtetési mód az előreláncolás: ha a következtető mechanizmus talál olyan szabályt, melynek feltételeit az adatbázisban tárolt tények kielégítik, akkor végrehajtja a szabályt, aminek hatására az adatbázis módosul. Mindez ciklikusan addig folytatódik, míg van olyan szabály, amely tüzelhet. A tüzelési ciklusnak három fő lépése van: (1) mintaillesztés, (2) választás, és (3) tüzelés (lásd 2. ábra).

A mintaillesztés kijelöli a szabálybázisban tárolt mindazon szabályok példányait, melyek feltételei illeszkednek az adatbázis pillanatnyi tartalmához. Ekkor nem pusztán a feltételek logikai ellenőrzése, hanem a feltételek közt szereplő változók minden lehetséges módon való lekötése történik.



2. ábra: A szabály-alapú következtetés ciklusa

Egy szabály többféle módon is illeszkedhet az adatbázisra, ezért valójában nem a szabályok tüzelnek, hanem azok példányai. Mivel azonban az adatbázisbeli változások lokálisak, nem érdemes minden ciklusban az összes szabály feltételeit újra és újra ellenőrizni (nagy tudásbázis esetén ezt nem is lehetne megtenni.) A részben teljesült feltételek folyamatos nyilván-tartására szolgál a szabályalapú következtető rendszerek egy belső, ún. RETE minta-illesztő mechanizmusa.

Példaként tekintsük az alábbi egyszerű ütemezési szabályt (a változók azonosítója ?-vel kezdődik, a feltétel és az akció oldalakat a  $\Rightarrow$  jel választja el egymástól, s a tények hozzáadása az add, törlése a del utasítások hatására történik):

```

(defrule tégy-munkadarabot-gépre
  ?f1: (gép ?G szabad)
  ?f2: (munkadarab ?M rendelkezésre-áll)
  (not (munkadarab ?M kész))
  (igényli ?M ?G)
  =>
  (add
    (munkadarab ?M gépen ?G)
    (gép ?G foglalt))
  (del ?f1 ?f2)
)
  
```



Ha az adatbázis – egyebek közt – tartalmazza azon adatokat, hogy (gép G3 szabad), (munkadarab M21 rendelkezésre-áll), és (igényli M21 G3), ám nem tartalmazza azt, hogy (munkadarab M21 kész), akkor a szabály illeszkedhet, és eközben a ?G változó felveszi a G3 értéket, míg ?M változó M21-et. Az adatbázis módosítása ezen változólekötések mellet történik.

Választáskor el kell dönteni, hogy a mintaillesztés során kijelölt szabály példányok közül melyik tüzeljen. Az esetleges konfliktus feloldására többféle ún. vezérlési stratégia használatos: az a szabály példány tüzel, amelyik a legfrissebb tényre illeszkedik, a legspecifikusabb (azaz a legtöbb feltételt tartalmazza), vagy amennyiben a szabályokhoz prioritás rendelhető, amelyeknek legnagyobb a prioritása. A tüzelés egy szabály adott – teljesült – feltételek mellett való végrehajtása, lekötött változók mellett. Ez módosíthatja az adatbázist, illetve mellékhatásokat eredményezhet. Hangsúlyozni kell, hogy arra nincsen semmiféle garancia, hogy egy feltételeiben kielégített szabály biztosan tüzeljen, hiszen mindig akadhat más szabály példány, amely előnyben részesül a választáskor, ill., amely hatásával elrontja a már kielégített feltételeket.

Az adatvezérelt következtetésen túl lehetőség van ún. cél-vezérelt, avagy hátraláncoló következtetésre is. A hátraláncolás egy adott célból indul ki és megvizsgálja, hogy a rendelkezésre álló tények alapján a cél igazolható-e. A mechanizmus először kiválaszt egy célt, majd egy szabályt, amelynek akció oldalán a cél állítása szerepel. Ez a cél kiterjesztése, amit általában több módon is meg lehet tenni. Ezután ellenőrzi, hogy az adatbázis kielégíti-e ezen szabály feltételeit; azokat a feltételeket, melyekre nincs illeszthető tényállítás, az igazolandó célokhoz csatolja. Ez a folyamat mindaddig tart, míg előáll egy megoldás, avagy a következtetés a célokat minden lehetséges módon ki nem terjesztette. Vegyes, mind előre-, mind hátraláncoló üzemmódban, ha a mintaillesztés egy bizonyos tény hiánya miatt nem sikerül, akkor ezt a tényt célként lehet kitűzni és hátraláncoló szabályok segítségével meg lehet próbálni az adatbázisban tárolt többi tényből levezetni.

A feltétel oldalon szereplő változók által felvehető értékeket külső eljárások is korlátozhatják; ezáltal a feltételek leírására szolgáló nyelv jelentősen gazdagodhat.

A szabálybázis sok esetben strukturálható: szabálycsoportok definiálhatók, melyek tetszés szerint ki- és bekapcsolhatók. A kikapcsolt szabálycsoportok nem vesznek részt a mintaillesztésben. A szabályokhoz prioritás rendelhető, amely nem is annyira a fontosságukat fejezi ki, hanem sokkal inkább végrehajtásuk sorrendjét határozza meg.



Nagyobb rendszerekben az adatbázis az objektum-orientált programozás eszközeivel strukturálható. Az ún. *frame-alapú* tudásreprezentáció és következtetés kihasználja, hogy gondolkodásunk sokszor tárgyakhoz, objektumokhoz, s főképpen azok közös jellemzőit megragadó fogalmakhoz, ún. osztályokhoz illetve prototípusokhoz kötődik, és lehetővé teszi osztályok, osztály hierarchiák és adott osztály(ok)ba tartozó példányok definiálását és ezen objektumok hatékony manipulálását.

! A *frame* olyan információ-tárolási alapegység, mely összefogja mindazon tulajdonságokat, melyek egyetlen objektumra jellemzők és azokat a relációkat, melyek ezen objektumot más objektumokhoz kötik. Mind a tulajdonságokat, mind a relációkat attributomokkal és a hozzájuk tartozó értékek párosaival lehet leírni. Az értékek megadása opcionális. A *frame* szolgálhat mind osztályok, mind példányok leírására. Ahogy egy objektumot többféle nézőpontból tekinthetünk, ugyanúgy megadhatjuk, hogy egy objektum egyszerre több osztály példánya is legyen. *Frame* rendszerekben alapvető következtetés az öröklődés: az egyes osztályok megöröklik a hierarchiában felettük álló osztályok attribútumait, és ha vannak, úgy az azokhoz rendelt értékeket. Ehhez hasonlóan, az egyes példányok is öröklik azon osztályok tulajdonságait és értékeit, amelyekbe beletartoznak. Ezeket az örökölt értékeket hívják alap, avagy *default* értékeknek.

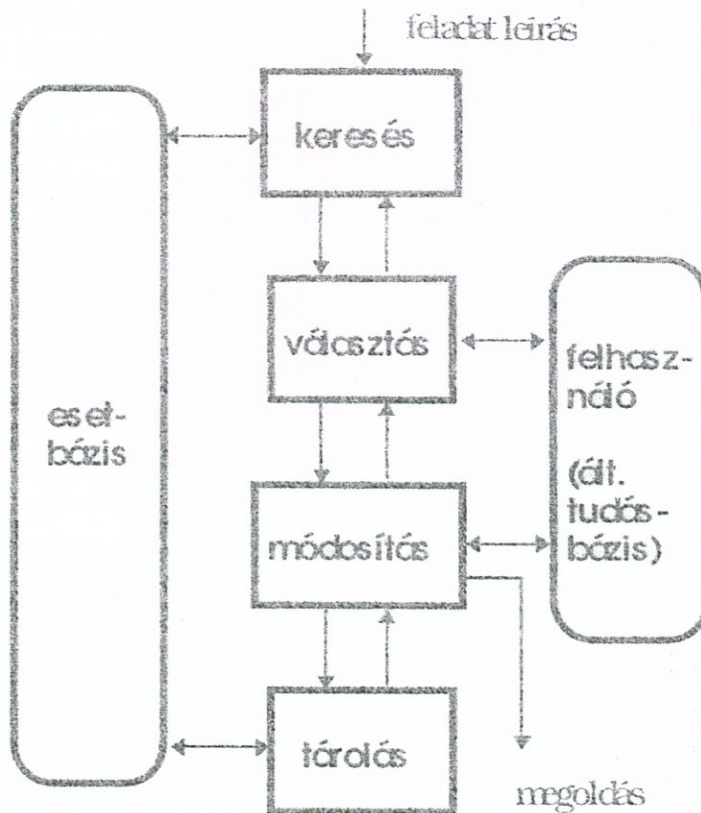
A gyakorlatban leginkább kedveltek a *frame*- és szabály-alapú reprezentációt integráló rendszerek, melyek adatbázisában az egyszerű tényállításokon túl különféle objektumokat leíró, *frame*-ekben összefogott tények is tárolhatók. Ekkor természetesen a szabályok feltételeinek megfogalmazásakor a *frame*-ekre és azok attribútum-érték párojaira is lehet utalni.

### 5.3. Eset-alapú következtetés

Korábban láttuk, hogy a *frame*-alapú reprezentáció közelebb hozta a világ logikai leírását az emberi gondolkodásmódhoz. Ám ez a reprezentáció is inkább olyan ismeretek leírására alkalmas, melyek nem változnak a használat során. Gondolkodásunkra sokkal inkább jellemző, hogy feladatok megoldásából tanulunk, hogy az így szerzett tapasztalatokra emlékszünk, hogy egyes ismereteink idővel finomulnak, míg mások elhalványulnak. Az igazi tapasztalat nehezen ragadható meg statikus formában, mert azt valójában állandóan változásban lévő epizódikus ismeretek szövevényes kapcsolata alkotja. Az emberi gondolkodásról alkotott efféle elképzeléseket veszi alapul az ún. eset-alapú következtetés (*case-based reasoning, CBR*), mely új problémákat



korábban megoldott feladatok megoldásainak visszakeresése és adaptálása révén próbál megoldani.



6. ábra: Az eset-alapú következtetés folyamata

Az eset-alapú következtetés sémája szinte meglehetősen egyszerű: tároljuk az eseteket – azaz a korábbi feladatok megoldásaiból származtatott epizódokat. Ezáltal létrehozunk egy esetbázist. A legegyszerűbb formában az esetek csak konkrét feladatokat és azok megoldásait tartalmazzák, de tárolhatók a megoldás előállításának lépései is, ill. a konkrét esetek általánosított változatai. Egy újabb megoldandó feladattal szembesülve keressük vissza azokat az eseteket, amelyekben a problémafelvetés a lehető legjobban hasonlít az aktuális problémához. Válasszuk ki a leginkább hasonló esetet, és annak megoldását használjuk fel újra. Ha nem kielégítő az így módon kapott megoldás, akkor próbáljuk meg újra felhasználni más hasonló esetek megoldásait, de végül, ha így sem kapunk megoldást, akkor egy korábbi feladat megoldását adaptáljuk az új körülmények között. Az ilyen módon kapott új megoldást – természetesen a hozzá tartozó feladatkitűzéssel együtt – tároljuk az esetbázisban. A következtetés kritikus pontjaiban komoly szerepet kaphat az ember is. Az eset-alapú következtetés folyamatát mutatja be a 6. ábra.



Az eset-alapú következtetés legnehezebb lépése a módosítás, azaz a régi megoldások adaptálása. Ha ezt a lépést akárcsak részben automatizálni akarjuk, szükség van a hagyományos logikai következtetési apparátus (leginkább a szabály-alapú következtetés) alkalmazására is. Egy új eset tárolásakor annak az esetbázisba való integrálását úgy kell megoldani, hogy jelentésbeli gazdagsága ne vesszen el, ám egyben könnyen visszakereshető is legyen. Ez az eset-alapú következtetés tanuló komponense.

#### 5.4. Szakértő rendszerek fejlesztése

A szakértő rendszerek fejlesztése két eltérő háttérű szakember (vagy csoport) szoros együttműködését igényli: a szakértőt és a tudásmérnököt (*knowledge engineer*). A szakértő az, aki az adott területen jártas és képes megoldani az ott felmerülő problémákat, míg a tudásmérnök biztosítja a megfelelő mesterséges intelligencia technikákat. A fejlesztési folyamat a szaktudás mérnöki megragadás – erre utal a *knowledge engineering* kifejezés.

A fejlesztés többszörösen visszacsatolt folyamat, ami magába foglalja akár a feladat újra definiálásának lehetőségét is. Nagy rendszerek építésekor nem ritka, hogy a fejlesztés során a tudásreprezentáció formáin is változtatni kell. Lényeges, hogy szakértő rendszerek fejlesztésekor mindig egy feladat megoldását kell nyújtani, nem pedig egyszerűen csak egy új szoftver technológiát. Ezért a fejlesztés nagyobb erőfeszítést igényel a szakértőtől, mint a mesterséges intelligencia ismereteket szolgáltató tudásmérnök részéről: a szakértő az, akinek teljes eddigi gyakorlatát újra kell vizsgálnia s nemritkán módosítania, és ő az, akire a verifikáció terhe szinte egészében ráhárul.

A tudásbeszerzés (*knowledge acquisition*) támogatására sokféle eszközt fejlesztettek ki: ezek jobbra általános protokollok, melyek segítségével a területre jellemző fogalmakra és szabályokra rá lehet kérdezni. Mivel egy szakértő tudása korábbi feladatok megoldása során szerzett tapasztalataiból származik, az epizodikus ismeretek jelentősége legalább akkora, mint az azokból kivonható, szabályszerű, általánosított tudásé. Ezért nő az eset-alapú rendszerek jelentősége, melyek módot adnak arra, hogy az elsődleges tapasztalatokat – konkrét feladatokat, megoldásaikat és a megoldások körülményeit – reprezentáljuk és a következtetés során felhasználjuk. Az elsődleges tapasztalatokra való rákérdezés nagyban könnyíti és gyorsítja a tudásbeszerzés egyébként igen nehézkes, hosszadalmas feladatát.

Szakértő rendszerek fejlesztésekor igen nehéz meghatározni azt a határt, ameddig a rendszerek kompetenciája terjed. A szabály-alapú rendszereknek



nincsen emlékezetük arról, hogy milyen feladatokkal találkoztak már. Így ugyanazt az utat járják be igen hasonló feladatok megoldásakor – s ez a hatékony működés ellen szól -, de ami ennél is rosszabb, hogy ugyanazon hibákat követik el újra és újra. Az emlékezet hiányából adódik, hogy a szabály-alapú rendszerek törékenyek: hirtelen működésképtelenné válnak olyan helyzetekben, amelyek kezelésére nincsen alkalmas szabályuk. Ilyenkor “leesnek a tudás sziklájáról”. Az eset-alapú és ideghálós rendszerek ezzel szemben egy adott feladattal valami távoli hasonlóságot mutató esetet mindig találhatnak, s a távolság még arra is figyelmeztet, hogy óvatosan kell bánnunk a javasolt megoldással. Ilyen értelemben tehát ezek a rendszerek robusztusak.

Elterjedt az a nézet, hogy egy kisebb tudás-alapú prototípus rendszer, ha egyszer már létrejött, újabb tudásdarabkák hozzáadásával nagy rendszerré bővíthető. Ez igaz az eset-alapú rendszereknél, feltéve, hogy biztosítva van a hatékony memória-hozzáférés lehetősége. A szabály-alapú rendszerek esetében a tapasztalatok szerint ennek éppen az ellenkezője igaz, mert a rendszer nagyléptékű bővítésekor mind nehezebb áttekinteni és kézben tartani a szabályok kölcsönhatásait. A bővítést nehezíti, hogy mivel nincs nyílt vezérlési struktúra, a vezérlés is egyes szabályok kölcsönhatásából adódik.

#### 5.4.1. Szakértő keretrendszerek

A szakértő rendszerek fejlesztését nagyban megkönnyítik az ún. szakértő keretrendszerek (*expert system shell*), melyek egy általános célú szerszámkészletet és egy annak megfelelő fejlesztő környezetet biztosítanak. Így a rendszerek fejlesztői az eszközkészítés helyett nyomban a megoldandó feladatra összpontosíthatnak. A korszerű keretrendszerek a következő szolgáltatásokat nyújtják:

- Integrált szabály- és frame-alapú tudásreprezentáció és következtetés. Ma már a legnagyobb rendszerek magukba foglalják az eset-alapú reprezentáció és következtetés eszközeit is.
- Integrált fejlesztő környezet, amely támogatja a tudásbeszerzést, magyarázatok szolgáltatását, karbantartást.
- Grafikus felhasználói interface.
- Adatbázis interface minden széles körben használatos adatbázishoz.
- Illesztés elterjedt hardware platformokhoz, és azok operációs rendszereihez. Ez utóbbi két szolgáltatás teszi lehetővé a más rendszerekkel való zökkenőmentes együttműködést, az összetett, esetleg más rendszerekbe is beágyazott mesterséges intelligencia rendszerek fejlesztését.





A mesterséges intelligencia alkalmazásoknak újabb lendületet adott, hogy a keretrendszereket immár hagyományos platformokon és a leginkább elterjedt operációs rendszerek alatt lehetett futtatni. A ma működő szakértő rendszerek mintegy fele valamilyen keretrendszer alkalmazásával valósult meg.

A szakértő keretrendszerek piaca meglehetősen széles, ahol mind az olcsó (pár száz dolláros), mind pedig az igen nagyigényű (néhány tízezer dollár nagyságrendű) rendszerek kaphatók. A gyorsan változó kínálatban is akad néhány stabil, kiemelkedő termék. Ezek közül most azokat ismertetjük, melyek kimondottan a technológiai menedzsment területén használhatók.

Az egyik legtöbb szolgáltatást nyújtó keretrendszer az ART\*Enterprise (gyártó: Brightware), mely lényegében objektum-orientált alkalmazások fejlesztésére alkalmas rendszer. Az ART\*Enterprise a 80-as évek óta tökéletesített ART (Automated Reasoning Tool) termékcsalád legújabb tagja. Támogatja a frame-, szabály- és eset-alapú reprezentációt és az ezeken való következtetést. Kétirányú kapcsolatot biztosít minden elterjedt adatbázis rendszerhez, platformtól függetlenül is, hálózati hozzáférés révén. Ez az adat-integráció azt jelenti, hogy a következtetések tárgyai a rendszerhez csak lazán csatolt adatbázisok, illetve dokumentumok is lehetnek. Az eset-alapú következtetésben használatos keresési mechanizmusa révén képes struktúrátlan dokumentumokban is keresni. Az ART\*Enterprise-nak saját objektum-orientált programnyelve van, de lehetővé teszi C++ kódú programok kihasználását és futtatását is. A rendszer egyik igen fontos eleme a grafikus felhasználói felület fejlesztését támogatja: lehetővé teszi, hogy bármely objektum reprezentációjának megalkotásával egyidejűleg szinte automatikusan keletkezzenek azok az eljárások, melyek az objektum prezentációját hivatottak kezelni. Az ART\*Enterprise többféle platformon is fut (Windows, OS/2, Unix/Motif, MVS). Legfrissebb változata már lehetővé teszi Interneten keresztül elérhető rendszerek létrehozását is. Legnagyobb ereje, hogy integráló szerepet tud betölteni nagyméretű, elosztott és heterogén információs rendszerekben. Az ART-IM (ART for Information Management) a termékcsalád egy korábbi generációjához tartozik: tudásreprezentációs és következtetési ereje hasonló, ám nem támogatja az adat-integrációt és a grafikus felhasználói felület fejlesztését.

Reprezentációs és következtetési erejét tekintve hasonló funkciókat, ám más fejlesztési stratégiát támogat a Haley Enterprise termékcsaládja, amely az ART\*Enterprise-al szemben elsősorban nem egy fejlesztői környezetet ad, hanem C++ nyelven írott programkönyvtárakat. A frame- és szabály-



alapú reprezentációt ill. előre- és hátraláncoló következtetést megvalósító alaprendszer az Eclipse. A Rete++ segítségével az Eclipse saját szabálynyelvén írott programok a következtető mechanizmussal együtt C++ kódú programokká fordíthatók. Az Easy Reasoner egy C++ programkönyvtár, mely az Eclipse-el együtt eset-alapú következtetést valósít meg. Az Easy Reasoner magába foglal egy indukciós programcsomagot is, amely lehetővé teszi, hogy az esetbázisban tárolt esetekhez döntési fán keresztül is hozzá lehessen férni. Összegezve, itt a hangsúly nem az integráló szerepen van, hanem azon, hogy rendkívül hatékony programokat lehessen írni, s hogy az intelligens szolgáltatások könnyen befoglalhatók legyenek nagyobb rendszerekbe.

Az eset-alapú következtetés eddig legsikeresebb eszköze a CBR2 rendszer (gyártó: Inference Corp.). Az esetek attribútum-érték párokkal adhatók meg. Az attribútumok a szokásos típusokon túl (numerikus, nominális, logikai) szabad szöveg jellegű értékeket is felvehetnek. A felhasználó megadhatja, hogy az egyes attribútumok milyen súlyt kapjanak a hasonló esetek visszakeresésékor. A visszakeresés a legközelebbi szomszéd módszere szerint történik (ekkor a szabad szövegű attribútumok közt igen erős string és alstring illesztő algoritmus keres hasonlóságot). Kereséskor a rendszer nem egyetlen megoldást javasol, hanem csökkenő távolság szerint rendezve felkínálja a leghasonlóbb eseteket. Egy alternatíva kiválasztásával előkerülnek mindazon releváns kérdések, melyekre az új probléma leírásában nem volt válasz; ha ezek valamelyikére a felhasználó válaszol, és újraindítja a keresést, akkor esetleg más tárolt esetekhez kerül közelebb az éppen megoldandó probléma leírása. A CBR2 a megtalált esetek adaptálására nem ad eszközt. Ha az új probléma nem hasonlít egyik ismert esethez sem, akkor meg kell ugyan oldani (a rendszer hatókörén kívül), de a megoldást, mint új esetet be lehet vinni az esetbázisba. A CBR2 interaktív fejlesztő rendszere lehetővé teszi a rendszer programozását számítástechnikában járatlan felhasználóknak is.

A CBR2 tehát kooperatív feladatmegoldást feltételez, a felhasználó aktív részvételével, de annak figyelmét a konkrét feladatban éppen fontosnak tűnő kérdésekre irányítva. Használatát az is könnyen megérti, akinek nincs különösebb gyakorlata a tudásalapú szoftverek alkalmazásában. Nincs szükség a tudásmérnök közvetítésére; azok kódolják az eseteket, akik közvetlen tapasztalattal rendelkeznek róluk, bár valójában az esetek összeválogatása, általánosítása, a releváns attribútumok és azok fontosságának megadása, az esetbázis strukturálása mind olyan feladatok, melyek sajátos szaktudást és fejlesztési módszert igényelnek. A CBR2 tanácsadást igénylő feladatok el-



látásában (mint pl. help-desk szolgáltatások, diagnosztika) immár bizonyítottan kiváltképp alkalmas. Nem alkalmazható azonban, legalábbis mint egyedüli eszköz, bonyolult tervezési, sorrendtervezési, ütemezési, s általában optimalizációs feladatok megoldásában.

A különféle rendszerekkel részletesen is meg lehet ismerkedni a világhálózaton keresztül: egyes programokat rögtön ki lehet próbálni, mások letölthetők és próbaideig használhatók, esetleg kézikönyveik böngészhetők. Az alapos előzetes tájékozódás már csak azért is igen fontos, mert egy rossz korai döntés nehezen orvosolható: a széles körben elterjed hiedelmekkel szemben egyik rendszerből a másikba nehezen lehet csak áttérni. Ennek oka, hogy a keretrendszerek, noha alapvető reprezentációs és következtetési módszereik közösek, a megvalósítás részleteiben igen elütnek egymástól.

#### 5.4.2. *Fejlődési tendenciák*

A tudás-alapú technológia és ahogyan azt a gyakorlatban alkalmazták, megteremtette önmaga korlátait is. Alább összegezzük azokat a törekvéseket, melyek ezen korlátok felszámolására irányulnak.

Az eddigiekkel ellentétben, kiemelt szerepet kap a tanulás, amely egyrészt példákból való indukció révén támogathatja a tudásbeszerzést, másrészt működés közben tökéletesíthet egy rendszert. Szakértő (főként szabály-alapú) rendszerek fejlesztésekor el kell tekinteni attól a tényről, hogy nincs olyan szakember, aki ne tapasztalatok révén tenne szert szaktudására; hogy a szakértő megszűnik azzá lenni, mielőst csak arra képes, hogy vakon alkalmazza szabályait, s lemond a tanulás lehetőségéről. Ez az elhanyagolás visszaüthet, ha az alkalmazási terület nem marad változatlan. Tanulásra leginkább, minden korlátozásaikkal együtt is, az eset-alapú szakértő rendszerek valamint a neurális hálók<sup>1</sup> képesek.

A szakértő rendszerek további terjedésének – és általában a mesterséges intelligencia sikeres alkalmazásának – immár komoly akadálya, hogy nincsenek más mérnöki tudományokhoz hasonlóan olyan szabványos építőelemek, amelyek felhasználásával intelligens rendszerek építhetők. Ne feledjük, a keretrendszerek csak szerszámokat adnak, de maguk üresek – itt most azonban éppen a tartalomról van szó. Szükség van szabványos, bárki által használható tudásblokkokra, reprezentációs modulokra, és ezek könyvtáaira

<sup>1</sup> Lásd később



Változik a szakértő rendszerek szerepe. Míg korábban az elsődleges cél egy ember által ellátott funkció kiváltása volt, ma egyre inkább az a cél, hogy a rendszer képessé tegye a felhasználót olyan funkciók betöltésére is, amire annak addig nem volt módja. Tehát nem a szakértőt helyettesítő rendszerek születnek, hanem asszisztens rendszerek szakértők számára. A sikeres szakértő rendszerek befoglalt rendszerek: egy nagyobb rendszer komponenseként működnek. Meglehetősen, a mesterséges intelligencia módszerek a közvetlen felhasználás szintjén meg sem jelennek, hanem pusztán prototípusok fejlesztése során, rosszul strukturált feladatok feltárásában kapnak szerepet.

Végül, a nemzetközi számítógéphálózat megteremtette a sokak által igénybe vehető szakértő szolgáltatások lehetőségét. Ma már egyszerű, bárki által használható technikája van annak, hogy egy nagy, szerver gépen futó szakértő rendszer (*knowledge server*) egy kliens gépen futó felhasználói felületen keresztül bárhonnan a világból elérhető legyen. Ez a megoldás platform független, egyszerűsíti a szakértő rendszer karbantartását és biztosítja annak jobb kihasználtságát.

## ELŐADÁS

A mesterséges intelligencia alapjai



### **Tudásreprezentáció és a következtetés módszerei**

- Frame alapú következtetés
- Szabály alapú következtetés
- Eset alapú következtetés

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Frame alapú tudásreprezentáció

-  Az emberi gondolkodás (emlékezet) általában tárgyakhoz, objektumokhoz kötődik.
-  Az emberi gondolkodás összekapcsolja az objektumot annak jellemző tulajdonságaival, viselkedésével, környezetével.

FRAME - információ tárolási alapegység, összefoglalja mindazon tulajdonságokat, amelyek egy objektumra jellemzők, és azokat a relációkat, melyek ezt az objektumot más objektumokkal összekapcsolják.



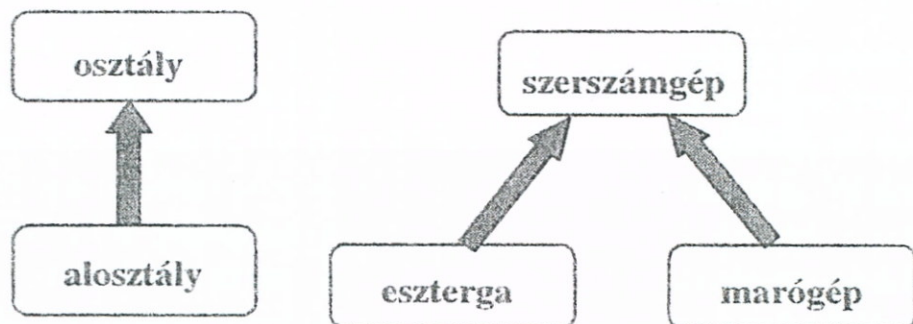
Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Alapelemek

Osztályok - Alosztályok - Példányok → hierarchikus felépítés

Példa: osztály - alosztály

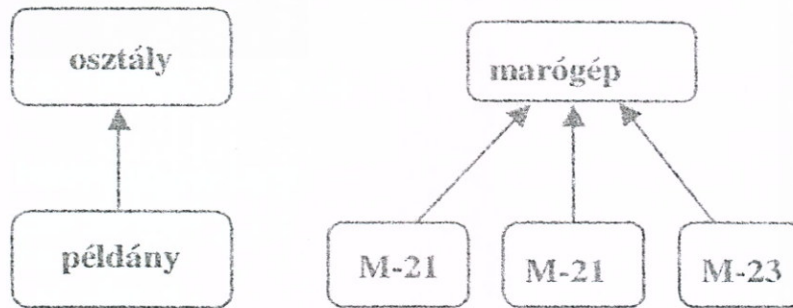


Dr. Szalay Tibor



A mesterséges intelligencia alapjai

Példa: *alosztály - példány*



- ➔ Magukban hordozzák a tulajdonságokat és relációkat
- ➔ Alapvető kapcsolatuk az öröklődés (tulajdonságok relációk öröklődnek a hierarchiában "felettük" (előttük) levőtől)
- ➔ Többszörös (több helyről való) öröklődés - több kapcsolat  
- több szint

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Technikai megvalósítás: *frame*

egyedi azonosító

*slot*: tulajdonság vagy reláció

speciális relációk

- osztály - alosztály (*is-a*)

- osztály - példány (*instance-of*)

érték

- *default*: örökölt értékek

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Példa

Osztály	(defframe macska
tulajdonság	(szín)
reláció	(megeszi (egér madár)))
Alosztály	(defframe házimacska
hierarchia	(is-a macska)
tulajdonság	(név)
reláció	(megeszi (konzerv madár)))
Példány	(defframe Mirci
hierarchia	(instance-of házimacska)
tulajdonság	(név Mirci)
reláció	(megeszi (konzerv kanári)))

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Kapcsolat a logikával (szemantika)

osztály - alosztály (is-a) reláció    *univerzális kvantálás*

$$\forall x \text{ házimacska}(x) \Rightarrow \text{macska}(x)$$

osztály - példány (instance-of) reláció

*állítás egy egyedi objektum adott osztályba tartozásáról*  
házimacska(Mir)

slot - érték párok    *nem egyértelmű a szemantika!*

$$\forall x \text{ házimacska}(x) \Rightarrow \exists y \text{ madár}(y) \wedge \text{megeszi}(x,y)$$

avagy

$$\forall x,y \text{ házimacska}(x) \wedge \text{madár}(y) \Rightarrow \text{megeszi}(x,y)$$

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

### A *frame*-alapú reprezentáció előnyei

- hétköznapi gondolkodáshoz illő reprezentáció
- hatékony következtetés az osztályok és az objektum példányok tulajdonságairól
- implementáció: objektum-orientált programozás
- Nincs szükség következtető eljárásra, az adatstruktúra hordozza a következtetést

Szakértő rendszerekben gyakran alkalmazott reprezentációs módszer

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Problémák - 1

#### Konfliktus öröklődéskor

*default* értékek és több helyről való öröklődés  
Honnan vegyük az értéket?

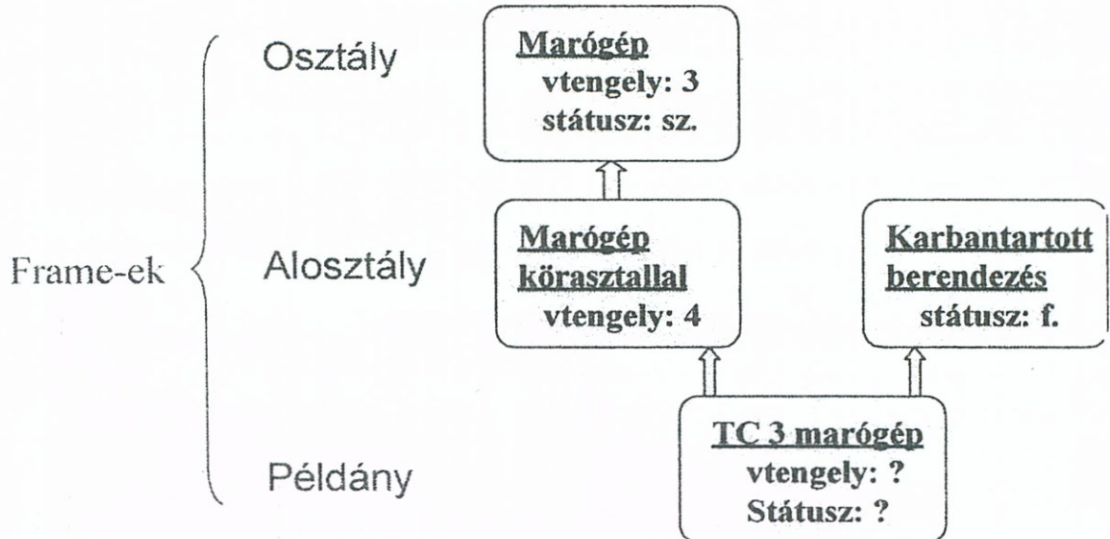
#### Példa:

- (1) A marógépek vezérelt tengelyeinek száma általában 3, és státusza szabad.
- (2) A körasztallal ellátott marógép is marógép.
- (3) A körasztallal ellátott marógép vezérelt tengelyeinek száma 4.
- 4) A karbantartott berendezések státusza foglalt.
- 5) M-33 egy karbantartás alatt álló, körasztallal ellátott marógép.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Reprezentáció



- konfliktus egy öröklődési út mentén
- konfliktus több út mentén

Dr. Szalay Tibor

A tudásreprezentáció alkalmazott módszerei

### Problémák - 2

#### Nem-monoton következtetés

monoton következtetés (I—)

Új tény hozzáadásával egy korábbi konklúzió nem válik hamissá (az ismert állítások halmaza monoton nő).

nem-monoton következtetés

Bármit állítunk, megfordítható az igazsága. *Default* értékek esetén a monoton következtetés nem működik.

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

### Példa

**tudjuk, hogy**

$\forall x \text{ madár}(x) \Rightarrow \text{repül}(x)$   
 $\text{madár}(\text{Pipi})$

**tehát:**

$\text{repül}(\text{Pipi})$

**újabb ismereteink:**

$\forall x \text{ pingvin}(x) \Rightarrow \neg \text{repül}(x)$   
 $\forall x \text{ pingvin}(x) \Rightarrow \text{madár}(x)$   
 $\text{pingvin}(\text{Pipi})$

**mindebből az (is) következik:**

$\neg \text{repül}(\text{Pipi})$



Ellentmondás!

D. Szalay Tibor

A mesterséges intelligencia alapjai

### Metódusok - Démonok

A slot-okhoz nemcsak logikai értéket (tulajdonság, reláció), hanem számítási eljárások is rendelhetők.  $\Rightarrow$  Metódus

A metódus a frame manipulálásakor aktivizálódik:

- *if-needed* ha a frame értékét vesszük át
- *when-changed* ha egy slot értéke változik

A slot-okhoz hasonló öröklődési problémák

D. Szalay Tibor

A mesterséges intelligencia alapjai

## Szabály-alapú reprezentáció és következtetés

Korlátozottabb nyelvű, mint az elsőrendű logika

Következtetési mechanizmusa kötöttebb

Hatékony és egyértelmű következtetések

Az emberi döntések kialakulásához közeli reprezentáció

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Szabálybázis

Ha-akkor típusú szabályok (implikáció,  $\Rightarrow$ )

Kissé eltérő értelmezés

A következmény inkább javaslat egy akció végrehajtására, mintsem egy egyszerű logikai állítás.

(defrule <szabály-név>

<Bármikor, ha ezen feltételek fennállnak>

$\Rightarrow$

<akkor hajtsd végre ezeket az akciókat>)

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

## Adatbázis

pozitív elemi tényállítások egy halmaza

Rögzíti mindazon tényeket, amelyeket a probléma megoldásának pillanatában a világról ismerünk.

Zárt világot feltételez, tehát minden állítás, amelyik nincs az adatbázisban, hamisnak tekintendő

Dr. Székely Tibor

A mesterséges intelligencia alapjai

## Példa

...

(gép G2 szabad)

(gép G4 foglalt)

(munkadarab M17 rendelkezésre-áll)

(munkadarab M25 rendelkezésre-áll)

(munkadarab M65 gépen G4)

(munkadarab M66 kész)

(igényli M17 G1)

(igényli M17 G2)

(igényli M25 G2)

...

Dr. Székely Tibor



A mesterséges intelligencia alapjai

## Működés

Szabály: Feltétel  $\Rightarrow$  Akció

feltétel

- adatbázisbeli tényekre vonatkozik.
- logikai kapcsolatok (ÉS, VAGY), ill. tagadás
- változók, tény minták

akció

- elemi tények hozzáadása ill. törlése
- mellékhatások (input/output)

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Példa -1

(defrule munkadarab-elkészült

?f1  $\leftarrow$  (munkadarab ?MD gépen ?G)

?f2  $\leftarrow$  (gép ?G megmunkálást-befejezte)

$\Rightarrow$

(assert (munkadarab ?MD kész)

(gép ?G szabad))

(retract ?f1 ?f2)

)

A változók azonosítója ?-lel kezdődik.

Tények hozzáadása - assert

Tények törlése - retract

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

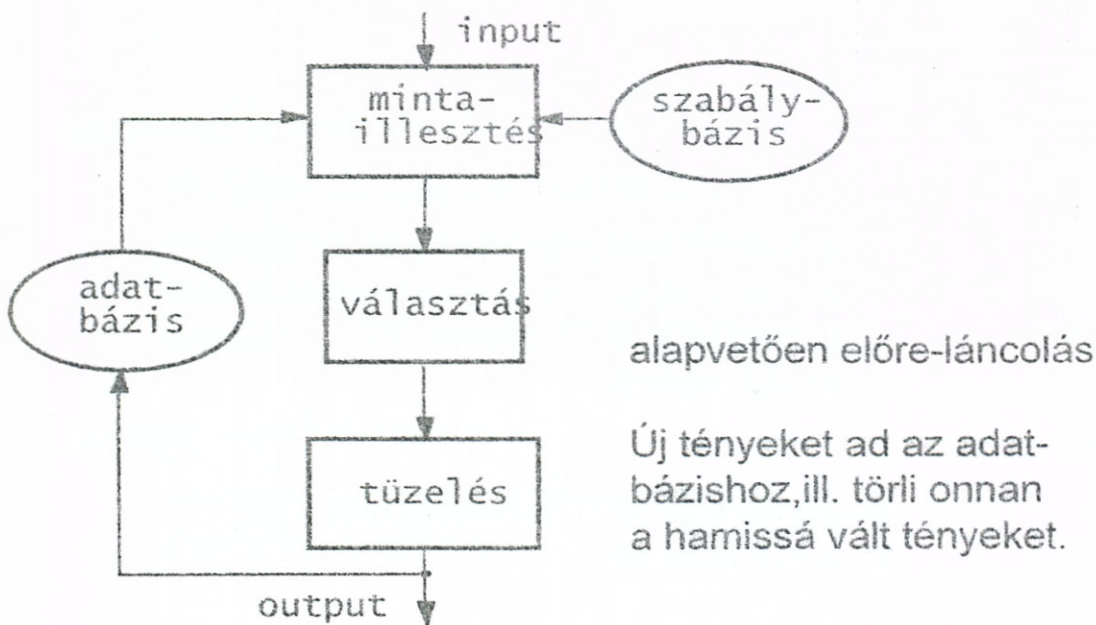
### Példa -2

```
(defrule tégy-új-munkadarabot-gépre
  ?f1 <- (gép ?G szabad)
  ?f2 <- (munkadarab ?MD rendelkezésre-áll)
  (not (munkadarab ?MD kész))
  (igényli ?MD ?G)
  =>
  (assert (munkadarab ?MD gépen ?G)
         (gép ?G foglalt))
  (retract ?f1 ?f2)
)
```

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Következtetési mechanizmus



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Elemei

### Mintaillesztés

feltétel oldali változók lekötése

Példa: tégy-új-munkadarabot-gépre

{?G/G2,?MD/M17} és {?G/G2,?MD/M25}

minden lehetséges módon (kimerítő keresés)

elvben unifikáció

a RETE mechanizmus

Szabálymemóriát hálóként reprezentálja

Definiál lekérdezéseket, egyesítéseket, cselekvéseket

Csak a változásokat frissíti (kicsi költség, hatékony mintaillesztés)

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Tüzelés

Módosítja az adatbázist és/vagy mellékhatásokat eredményez

Pl. ha a tégy-új-munkadarabot-gépre szabály  
{?G/G2,?MD/M17} illeszkedéssel tüzel, akkor

1. új tények: (munkadarab M17 gépen G2)  
(gép G2 foglalt)

2. törölt tények:

(munkadarab M17 rendelkezésre-áll)  
(gép G2 szabad)

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

## Választás

Bizonyos esetekben több szabály példány is tüzelhet,  
ilyenkor választás = konfliktus feloldás

Stratégiák: legfrisebb tényre illeszkedve  
legspezifikusabb tényre illeszkedve  
legnagyobb prioritású tényre illeszkedve

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Eset-alapú tudásreprezentáció és következtetés

Alapfeltevés: amilyen volt a múlt, olyan lesz a jövő is  
az "igazi" tapasztalat nehezen ragadható meg  
szabályok segítségével azt többé-kevésbé  
általánosított esetek szövevényes kapcsolata  
alkotja

Módszer: újra-felhasználás

Korábbi feladatok sikeres megoldásainak  
újr felhasználása.

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

### Alkalmazása

- A szakterület nehezen formalizálható szabályokkal
- Az alkalmazás átfogó szabályokat igényel
- A szakterület eleve eset alapú (orvosi diagnosztika, jog)
- Túl sok szabály kellene a szakterület leírásához
- Dinamikus szakterület az új problémák megoldásának gyors leírása szükséges

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Esetleírás

Probléma és annak leírása  
(attributum - érték párokkal)

Példa:

szabad szöveg

A lézernyomtató maszatos oldalakat nyomtat ki.

numerikus (fuzzy)

(kora=öreg); 8 és 10 év közé esik  
(nyomtatott-oldalak-száma=15739)

szimbolikus

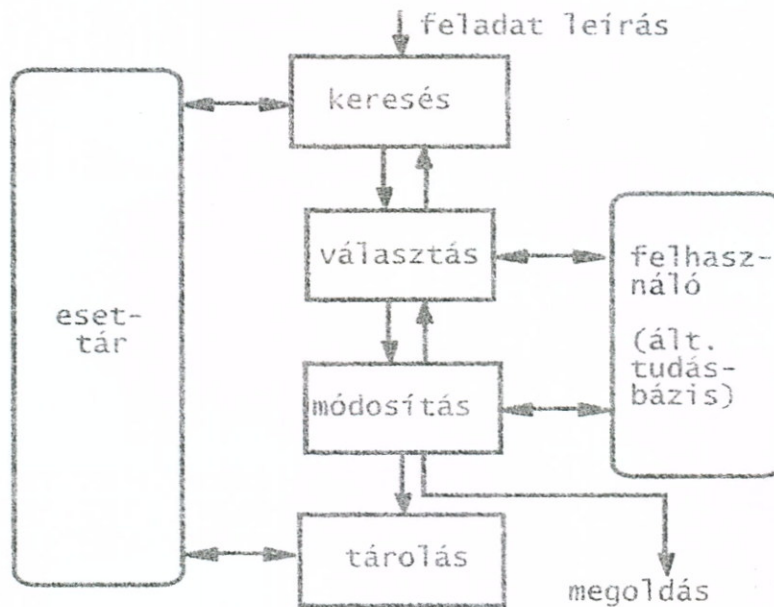
(típus=LaserWriter\_II)  
(gyártó=Canon)

Dr. Szalay Tibor



A. mesterséges intelligencia alapjai

### Működése



Dr. Szalay Tiber

A. mesterséges intelligencia alapjai

### Keresés - Választás

#### Eset illesztés

- hasonlóság
  - közelség/távolság
  - attribútum típusonként más és más módon mérhető
- súlyozás
  - egyetlen kombinált távolság
- irányított kikérdezés és kontextus-függő visszakeresés
- a siker kulcsa: jó indexelés

#### Választás

- Illeszkedő esetek közül a legalkalmasabb kiválasztása.

Dr. Szalay Tiber

A mesterséges intelligencia alapjai

## Módosítás - Tárolás

### Módosítás

korábbi megoldás új körülményekhez való adaptálása  
nagyon nehéz automatizálni

### Tárolás

új eset elhelyezése az eset-tárban (gazdag indexelés)  
egyféle tanulás

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Előnyök, hátrányok

egyszerű létrehozni

használat közben javul

de

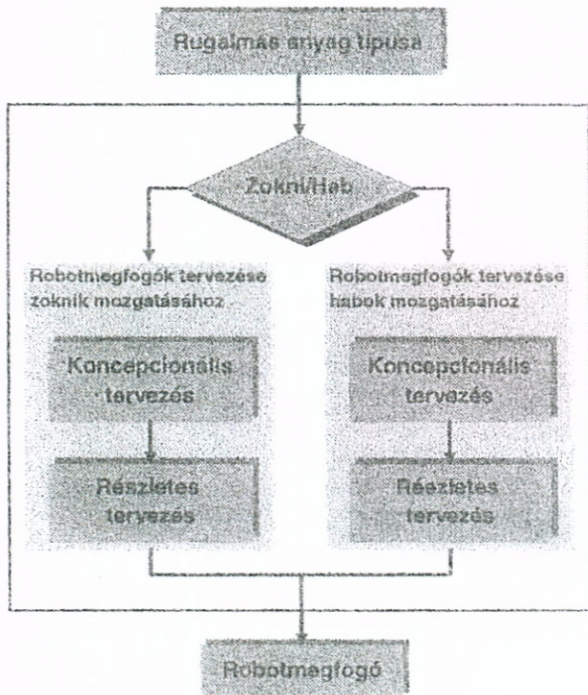
informális (nem egzakt) megoldás

csak az esetek által lefedett problémákat  
oldja meg

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

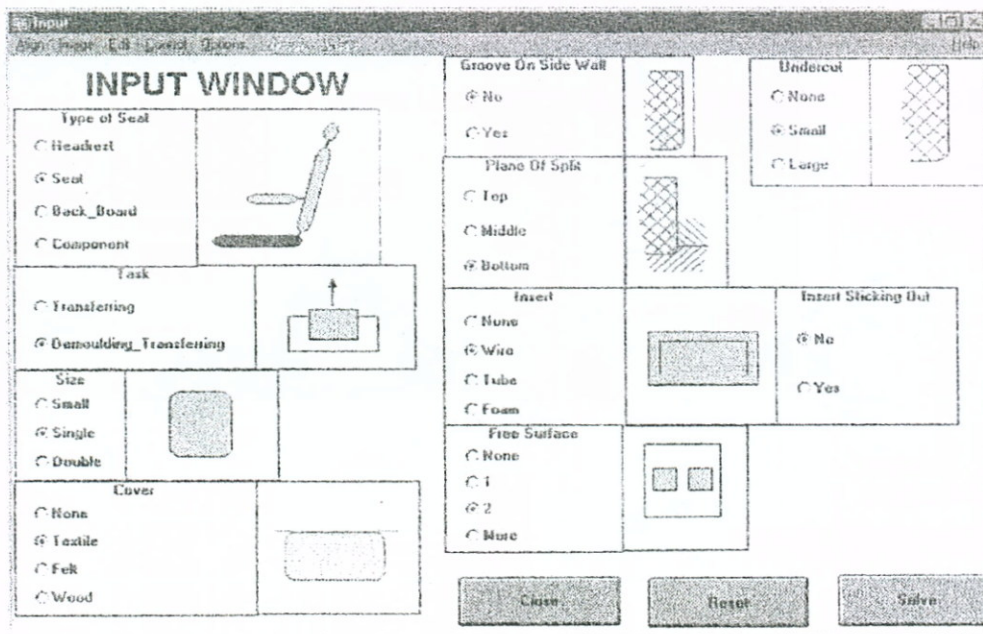


Eset alapú következtető rendszer robotmegfogók tervezéséhez

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

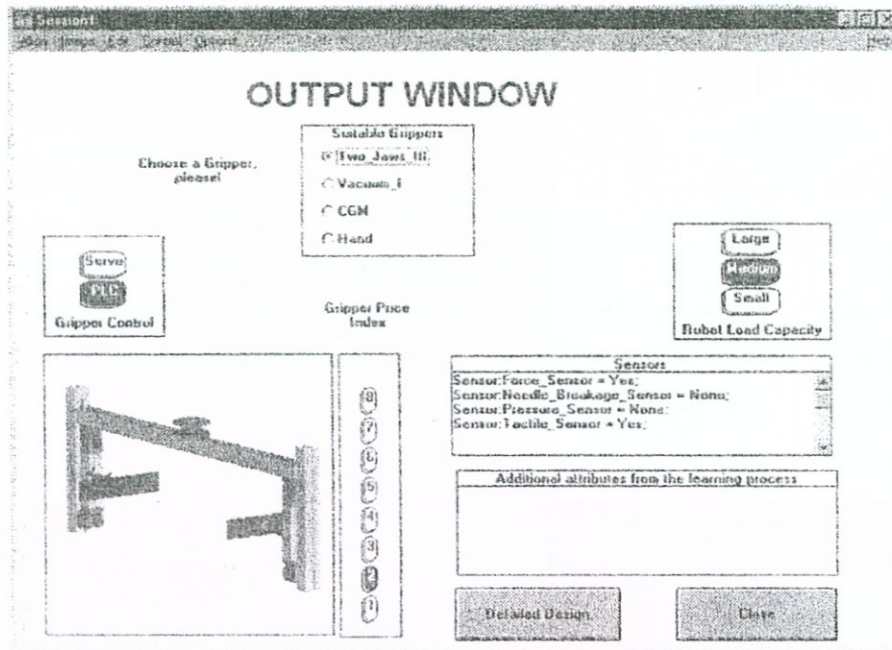
### Szakértői rendszer felhasználói felület



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

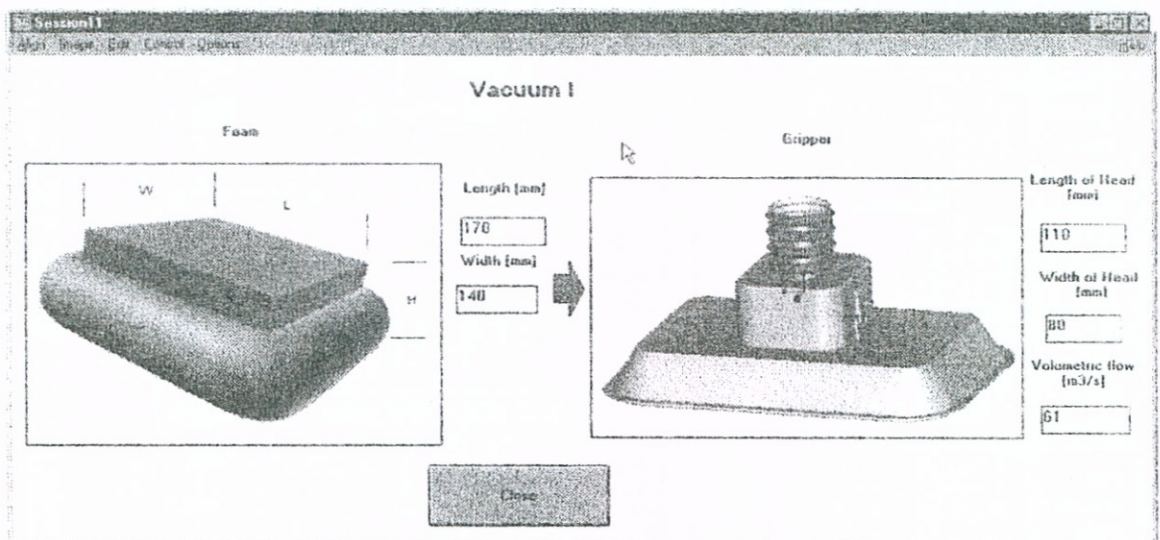
## Szakértői rendszer felhasználói felület



Dr. Szalay Tiber

A mesterséges intelligencia alapjai

## A vákuum megfogó részletes tervezése

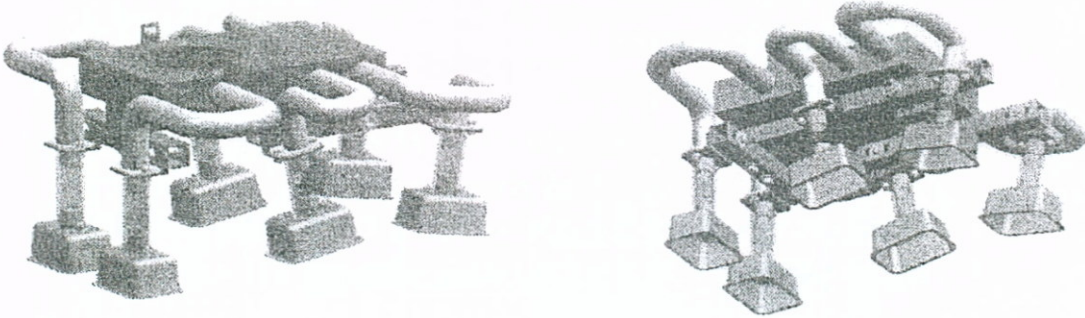


Dr. Szalay Tiber



A mesterséges intelligencia alapja:

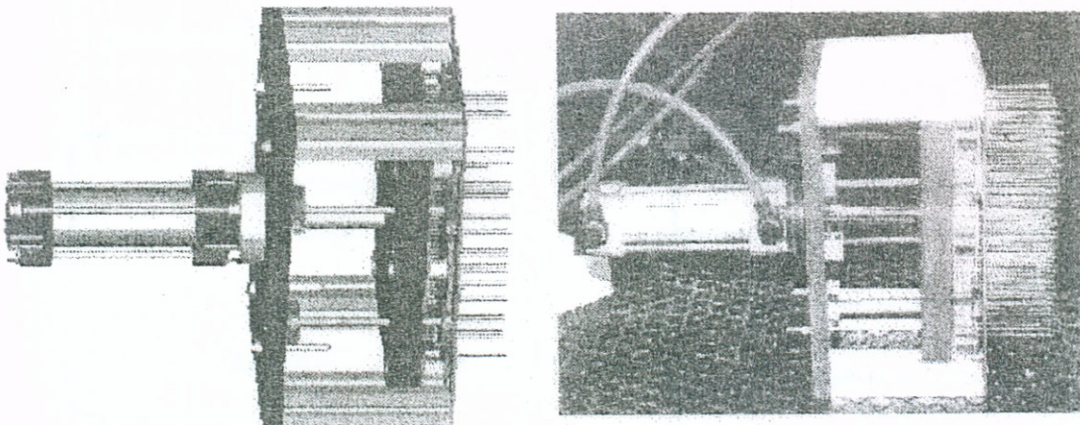
## Megfogó kialakítás



Dr. Szalay Tibor

A mesterséges intelligencia alapja:

## Tűmátrix megfogó



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Szakértő (tudásalapú) rendszerek

Keretrendszerek (*shell*)

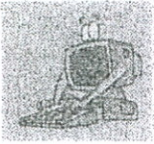
szerszámkészlet

üres rendszerek

- reprezentációs nyelv → tudásbázis
- következtető mechanizmus

-felhasználói interface

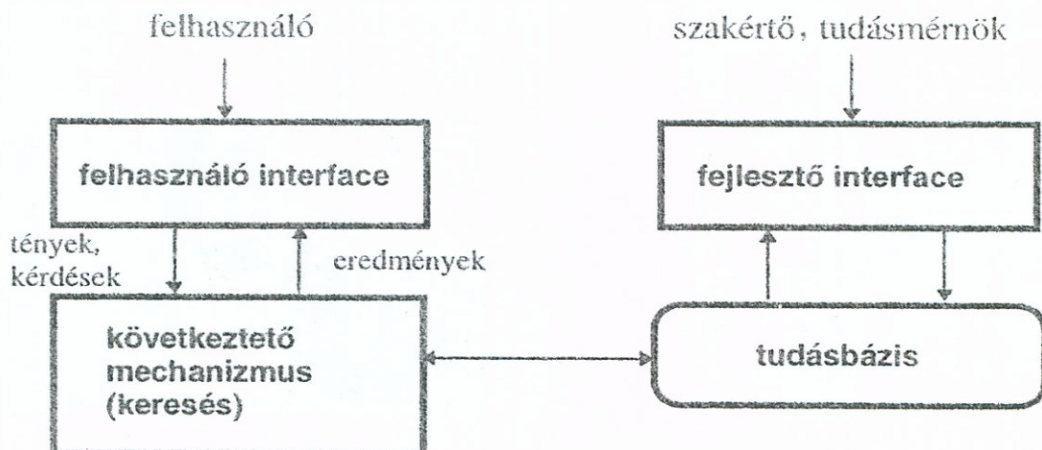
-fejlesztő környezet (leválasztható)



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## Felépítése



Dr. Szalay Tibor



A mesterséges intelligencia alapjai

## Szolgáltatások

integrált (szabály-, frame- és esetalapú)  
reprezentáció és következtetés

- fejlesztő környezet (tudásbeszerzés,  
magyarázat szolgáltatás, karbantartás)
- grafikus felhasználói interface
- adatbázis interface

többféle platformon (hardware, operációs rendszer)

## Példa

Doctus shell (magyar)



Dr. István Tóth





## 6. A GÉPI TANULÁS

A mesterséges intelligencia alapjai

### Gépi tanulás

#### Motiváció

az emberi tanulási folyamat modellezése

általános tanulási modellek felállítása

A tanulás szerepe a racionális viselkedésben.

tudásalapú rendszerek fejlesztése és tökéletesítése

- tudásbeszerzés támogatása
- törekenység kiküszöbölése

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Definíció

#### adott a

- feladatok egy osztálya (F)
- működés kiértékelésének kritériuma (K), és a
- működés során nyert eddigi tapasztalatok (T)



cél: a tapasztalatok (T) alapján változtassuk meg a rendszer működését oly módon, hogy az adott osztályba (F) eső feladatokat ezentúl jobban oldja meg. A kiértékelés az adott kritériumok (K) alapján történjen.

fontos: K a rendszeren kívülről való

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Példák**

kézi címzés felismerése

F: ismerd fel az irányítószámot kézzel címzett leveleken

K: helyesen felismert irányítószámok százalékos aránya

T: 100 000 kézzel írott cím, ismert irányítószámokkal

robot-jármű vezetése

F: vezesd a járművet az M7-es autópályán

K: sebesség/[1 + ütközések száma]<sup>10 000</sup>

T: 10 órányi videofelvétel szinkronban a vezetési akciókkal

Dr. Szalay Tócs

A mesterséges intelligencia alapjai

mikor menjünk teniszezni?

F: szabad ég alatt való teniszezésre alkalmas időpont választása

K: kedvező időjárási viszonyok mellett végigjátszott mérkőzések százalékos aránya

T: eddig milyen időjárási viszonyok mellett hogyan zajlottak a mérkőzések (végigjátszották vagy sem)

Dr. Szalay Tócs



### Tanulási módszerek

induktív (hasonlóságon-alapú) tanulás

- tanuló példákból való általánosítás
- indukciós elv
- Ockham borotvája **William of Ockham (1285-1349)**

„Entia non sunt multiplicanda praeter necessitatem”

„Az entitásokat nem szabad szükségtelenül szaporítani”

analitikus vagy deduktív (magyarázaton alapuló) tanulás

- meglévő ismeretek (tudásbázis) átszervezése
- miért tanulás mégis? más lesz a hatékonyság ...

nem-szimbólikus módszerek

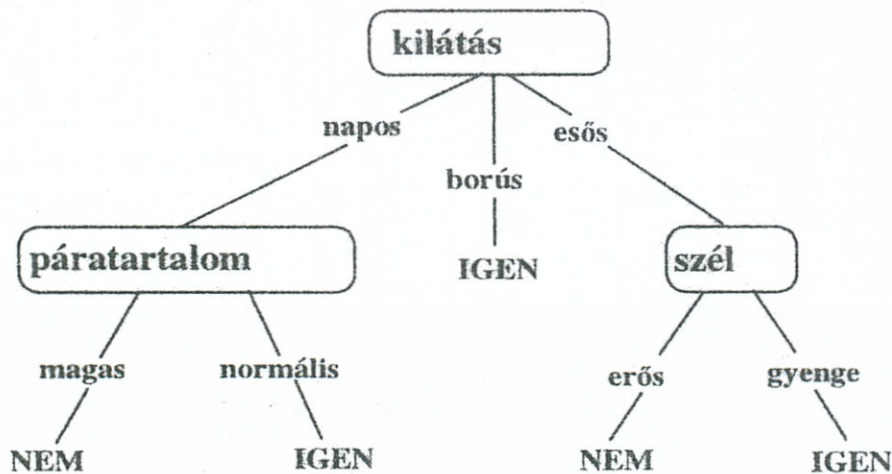
neurális hálók

Dr. Szalay Tibor

### 6.1. A döntési fa módszere

#### Induktív tanulás: döntési fa előállítása

Érdemes teniszezni?



Osztályozás egyes attribútumok értékeinek tesztelése alapján

Dr. Szalay Tibor



A mesterséges intelligencia alapjai

### Kapcsolat a logikai reprezentációval

jó a döntési fa

ha a lehető legkevesebb teszttel döntésre jutunk

utak a döntési fán = implikációk (szabályok)

$\text{érdemes-teniszezni} \leftarrow (\text{kilátás}=\text{napos}) \wedge$   
 $(\text{páratartalom}=\text{normális})$

döntési fa = egy fogalom diszjunktív definíciója

$\text{érdemes-teniszezni} \leftarrow$   
 $(\text{kilátás}=\text{napos}) \wedge (\text{páratartalom}=\text{normális}) \vee$   
 $(\text{kilátás}=\text{borús}) \vee (\text{kilátás}=\text{esős}) \wedge (\text{szél}=\text{gyenge})$

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Tanuló példák

kilátás	hőmér- séklet	páratar- talom	szél	érdemes teniszezni
napos	meleg	magas	gyenge	nem
napos	meleg	magas	erős	nem
borús	meleg	magas	gyenge	igen
esős	enyhe	magas	gyenge	igen
esős	hűvös	normális	gyenge	igen
esős	hűvös	normális	erős	nem
borús	hűvös	normális	erős	igen
napos	enyhe	magas	gyenge	nem
napos	hűvös	normális	gyenge	igen
esős	enyhe	normális	gyenge	igen
napos	enyhe	normális	erős	igen
borús	enyhe	magas	erős	igen
borús	meleg	normális	gyenge	igen
esős	enyhe	magas	erős	nem

egy tanulandó fogalom  
(osztály) példányai

reprezentáció:  
 <attribútum, érték>  
 párok  
 + osztályba sorolás

Dr. Szeley Tibor



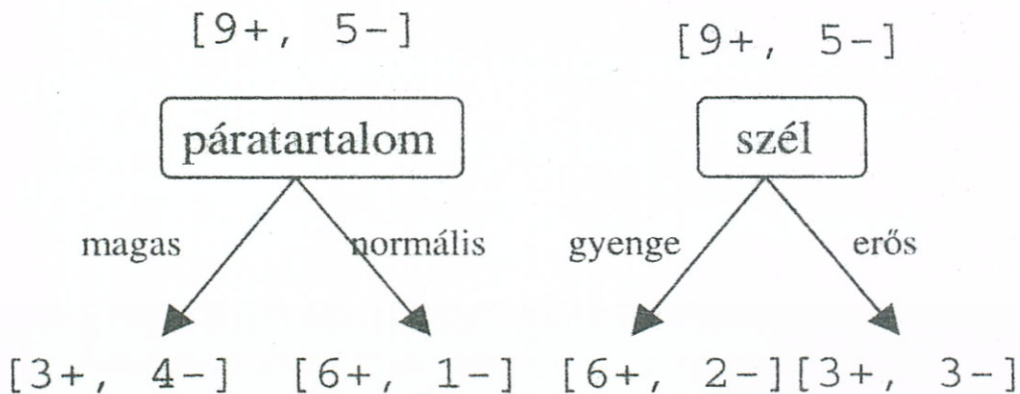
Döntési fa tanulása

- 1) Válaszd ki a "legjobb" attribútumot.
- 2) E csomópontból kiindulva bővítsd ki a fát az attribútum minden lehetséges értéke szerint.
- 3) Az értékek szerint csoportosítsd a példákat a levelekhez.
- 4) Minden levélre egyenként  
ha az csupa azonos osztályozású példát tartalmaz,  
akkor rendeld hozzá ezt az osztályozást és állj le;  
egyébként ismételd meg rá az 1) -4) lépéseket.

Módszer: felülről-lefelé való indukció

Dr. Szalay Tibor

**Melyik a legjobb attribútum?  
Melyik osztályoz jobban?**



Milyen információt nyerünk, ha A attribútumot teszteljük?

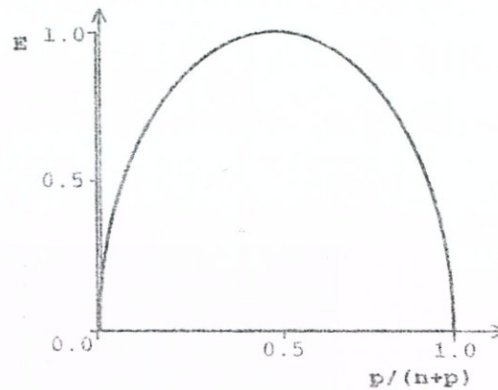
Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### entrópia $E(I)$

$I$ : a tanuló példányok halmaza,  
amely  $p$  pozitív és  $n$  negatív példát tartalmaz.

$$E(I) = - [p/(p+n)] \cdot \log p/(p+n) - [n/(p+n)] \cdot \log n/(p+n)$$



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Nyereség

Cél: legyen maximális

$\text{Gain}(A,I)$ : az entrópia várható csökkenése,  
ha  $A$ -t választjuk

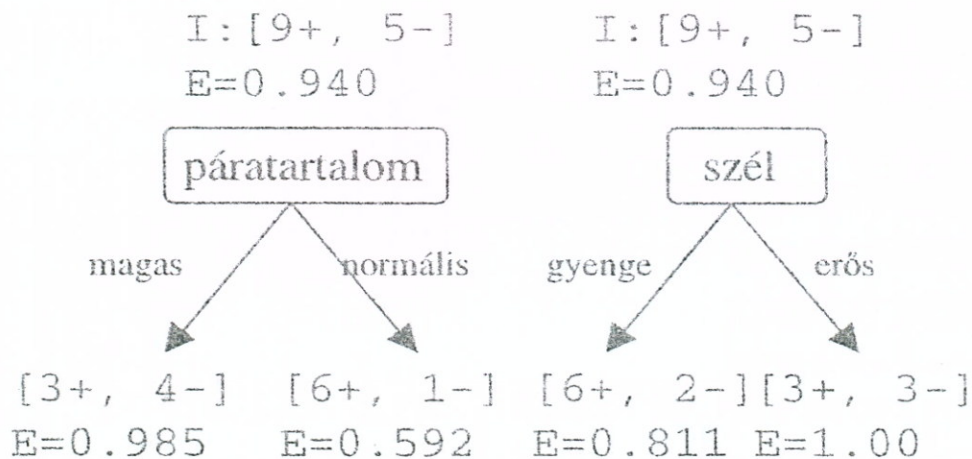
$$\text{Gain}(A,I) = E(I) - \sum_j \frac{p_j+n_j}{p+n} E(I)_j$$

Válasszuk azt az attributumot, amelyre  $\text{Gain}(A,I)$  maximális!

Dr. Szalay Tibor



## Példa



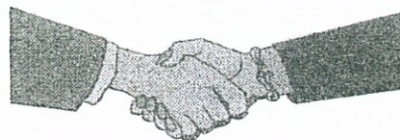
Dr. Szalay Tíme



$$\begin{aligned}
 \text{Gain}(I, \text{páratartalom}) &= 0.940 - (7/14) * 0.985 - \\
 & (7/14) * 0.592 \\
 &= 0.151
 \end{aligned}$$

$$\begin{aligned}
 \text{Gain}(I, \text{szél}) &= 0.940 - (8/14) * 0.811 - \\
 & (6/14) * 1.00 \\
 &= 0.048
 \end{aligned}$$

Azaz, ha a példákat a **páratartalom** szerint osztályozzuk, több információt nyerünk, mintha a **szél** szerint osztályoznánk.



Dr. Szalay Tíme

## 6.2. A legközelebbi szomszéd módszere

A mesterséges intelligencia alapjai

### Induktív tanulás: a legközelebbi szomszéd módszere

#### Memória-alapú tanulás

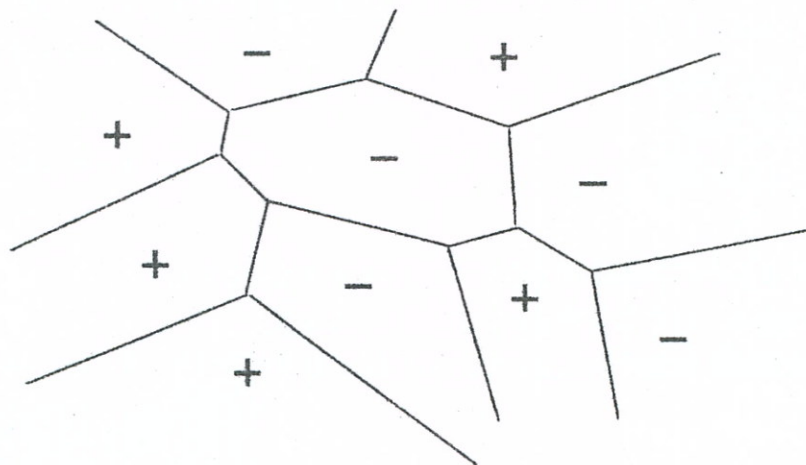
- definiálj egy távolság fogalmat (metrika)
- tárold el a tanuló példákat
- egy új feladat esetén keresd ki a legközelebbi szomszédot;
- az eredmény legyen ennek az osztályba sorolása

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

#### Tanulási folyamat

- 1) határok meghúzása
- 2) határok felszámolása



Dr. Szalay Tibor



## Távolság fogalom - k-legközelebbi szomszéd

Euklideszi távolság

Ha X példány reprezentációja:  $x_1, \dots, x_n$  vektor és

Y példány reprezentációja:  $y_1, \dots, y_n$

$$d(x, y) \equiv \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

Kiválasztja a tanuló mintából egy adott X példányhoz legközelebb eső k darab példányt

Dr. Szalay Tíme

## Példa

Valós értékű függvények megtanulása

Adott S a tanuló példák halmaza, és egy x példány esetén kívánjuk meghatározni  $f(x)$  függvény értékét. Keressük meg a k legközelebbi példányt ( $s_i \in S$ ,  $i=1, \dots, k$ ) a tanuló példák közül, ezek után

$$f(x) \equiv \frac{\sum_{i=1}^k f(s_i)}{k}$$

Dr. Szalay Tíme

A mesterséges intelligencia alapjai

## **Alkalmazás**

Eset alapú következtetés/visszakeresés

Tulajdonságok

a tanulás gyors és nem jár információ veszteséggel  
nehéz jó metrikát találni

A példa alapján történő, illetve eset alapú tanulást "lusta" tanulási módszernek nevezzük, mert nincs egy előre meghatározott globális célfüggvény, hanem minden példányra külön számítunk megfelelő célfüggvényt

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

## **Analitikus tanulás**

Tanulás dedukció révén

- igazságmegőrző lépések
- új, de a már ismert axiómákból eredő axiómák hozzáadása a tudásbázishoz

Probléma: hasznos szabályokat kell előállítani

Nő a hatékonyság?

- elágazási tényező
- felejtési is kellene

Dr. Szalay Tibor



### 6.3. A neurális hálók módszere

Élő szervezetek idegszöveti és az idegszervek specifikus idegi és nem specifikus általános szöveti elemekből épülnek fel. A specifikus ingerlékenységi és ingerületvezetési működésekben kizárólag a specifikus idegelemek, ún. *neuronok* vesznek részt. A neuronok rövidebb nyúlványait dendriteknek, a legfeljebb később elágazódó főnyúlványát neuritnek nevezzük. Az emberben akár 1 m-nél is hosszabb neurit, vagy idegnyúlvány más idegelemen, vagy egyéb szövetelemen ér véget. A neuronok közötti kapcsolódások a *szinapsziszok*.

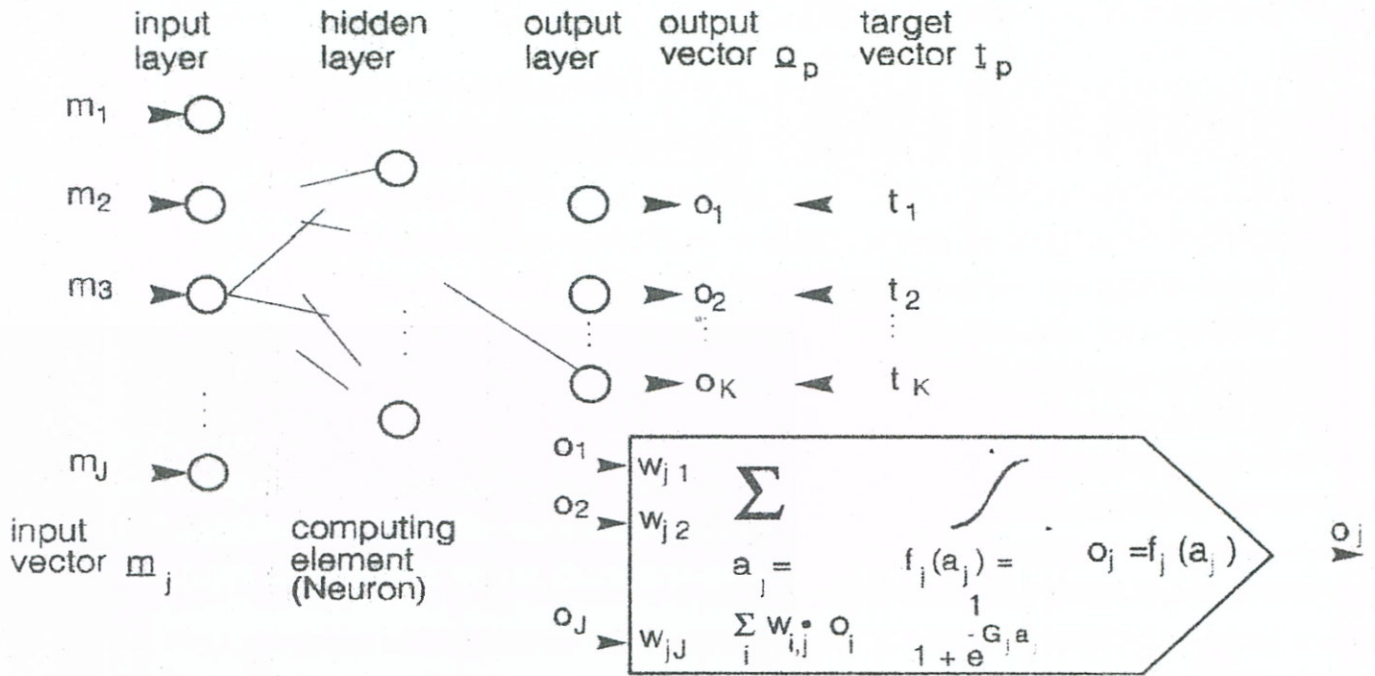
*A mesterséges neurális hálók (artificial neural networks, ANNs) "egyszerű, általában adaptív elemek sűrűn összekapcsolt hálózatai, és ezek hierarchikus szervezetei, amelyek a valós világ objektumaival olyan kölcsönhatásban állnak, mint a biológiai idegi rendszerek".*

A mesterséges neurális hálókra vonatkozó kutatások két fő csoportba oszthatók. A *biológiai modellezés* célja az élő idegrendszer struktúráinak és működésének tanulmányozása, hogy választ lehessen adni pl. a viselkedéssel kapcsolatos kérdésekre. A *technológiai modellezés* során a biológiai rendszerekben feltárt elveket szeretnénk új számítási technikák kifejlesztésében felhasználni. Itt olyan megoldások is létjogosultságot nyerhetnek, melyek neuróbiológiailag nem megalapozottak. A következőkben neurális hálók alatt mesterséges neurális hálókat értünk, míg a biológiai neurális hálók esetén feltüntetjük a megkülönböztető "biológiai" jelzőt.

A neurális hálókat a hálók topológiája, a "neuronok" karakterisztikája és az alkalmazott tanulási algoritmusok (lásd később) határozzák meg. Az összeköttetések elrendezése alapjaiban meghatározza a neurális hálók jellegét. Ebben az értelemben beszélhetünk *előrecsatolt (feed forward)* hálóról és olyan hálóról is, amelyek *visszacsatoló* hurkokat (*feedback loops*) is tartalmaznak. Más szemszögből megkülönböztethetünk *teljesen (fully)*, illetve *ritkán (sparsely connected)* csatolt hálókat. (Az első esetben a neuronok a háló minden más neuronjával összeköttetésben állnak, míg a második esetben csak néhány más neuronhoz vezet összeköttetés.) Rétegzett hálónál a neuronok úgy rendezhetők csoportokba, rétegekbe, hogy közbülső (rejtett) réteg esetén az egy rétegen belüli elemek kizárólag csak a szomszédos rétegekkel, kimeneti, illetve bemeneti rétegek esetén csak a szomszédos rétegekkel és a külvilággal állnak kapcsolatban.



A hálók neuronjait, másnéven csomópontjait belső *küszöbértékükkel* és *átviteli függvényükkel* (pl. bináris, digitális, analóg, lineáris, nemlineáris) jellemezhetjük. A 7. ábra a leggyakrabban használt előrecsatolt, rétegzett hálót mutatja analóg, nemlineáris átviteli függvényekkel rendelkező neuronokkal. A csomópontok a bemenetek súlyozott összegének a neuron küszöbértékével módosított összegét bocsátják egy szigmoid alakú nemlinearitás bemenetére.



7. ábra Tipikus előrecsatolt, rétegzett neurális háló analóg, nemlineáris átviteli függvényekkel rendelkező neuronokkal

### 6.3.1. Neurális modellek



A kutatók egységes véleménye szerint az egyik első, már megfelelően absztrakt neuron-modellt McCulloch és Pitts vezette be 1943-ban. A modelljükben leírt neuron aktivitása a bemenetére érkező jelek súlyozott összegétől függ. A kimeneti jel tipikusan a neuron-aktivitás nemlineáris függvénye. A McCulloch-Pitts neuron a kimenetnél teszi lehetővé egy küszöbérték szerepeltetését.

A mesterséges neurális hálók fejlődéstörténetében a következő főbb mérföldkövek emelhetők ki:

- 1958-ban Rosenblatt tanuló eljárást vezetett be a McCulloch-Pitts neuronra. *Perceptron* modelljében a súlyokat a háló aktuális és meg-



kívánt kimenete különbségéből számított hiba visszacsatolásán alapuló eljárással változtatta.

- 1960-ban Widrow és Hoff az egy kimenetű *adaline* (*adaptive linear neuron*) és a több kimenettel rendelkező *madaline* (*many adalines*) modelljeit írta le. Mindkét modellre a legkisebb négyzetek elvén alapuló tanulóeljárást fejlesztett ki.
- 1962-ben publikálta Rosenblatt *többszintű perceptronját*, egy valószínűségi alapon működő tanulóalgoritmussal együtt.
- 1969-ben Minsky és Papert rámutatott, hogy elemi perceptronok nem képesek a kizáró vagy művelet leképzésére. Közlésük hatására a neurális hálókkal kapcsolatos kutatások támogatása drasztikusan lecsökkent.
- Hopfield egyrétegű, teljesen csatolt bináris hálója új lendületet adott a kutatásoknak. A felügyelt tanulásra (lásd később) képes hálót sikerrel alkalmazták bonyolult optimalizálási feladatok megoldására.
- Az igazi áttörést a *back propagation* tanulási eljárás bevezetése jelentette 1986-ban. A Rummelhat, Hinton és Williams által leírt algoritmus a korábbi, legkisebb négyzetek elvén alapuló eljárás általánosítása többrétegű, nemlineáris átviteli függvényekkel rendelkező hálókra. (A mesterséges neurális háló tanulóalgoritmusaival foglalkozó fejezetben az eljárással bővebben foglalkozunk.)
- Nagy hatású volt Kohonen *self organising feature maps* elnevezésű hálója is, ahol a bemeneti neuronok közvetlen összeköttetésben állnak a kétdimenziós struktúrában elrendezett, sok lokális kapcsolattal rendelkező kimeneti réteg neuronjaival. A felügyelet nélküli, versenyztető tanulási fázis (lásd később) során az összeköttetések erősségét jelző súlyok olyan értékeket vesznek fel, hogy hasonló bemenetekre a kimeneti réteg topológiailag közeli neuronjai reagálnak.
- Carpenter és Grossberg adaptive resonance theory (ART, bináris inputokra ART1, folytonos bemenetekre ART2) hálói összetett struktúrák, melyek több elonyös tulajdonsággal rendelkeznek, például elég stabilak, hogy új információk hatására is megtartsák a korábbi tanulás fontos eredményeit, de emellett nyitottak újabb információk befogadására is.

Az elmúlt évtizedben a neurális háló sokasága jelent meg, itt csak a legfontosabbakat tudtuk felsorolni. További típusokkal kapcsolatban az irodalomra utalunk. Egy tényt azonban szükségesnek tartunk kiemelni, mégpedig a programozható neurális chipek megjelenését.



### 6.3.2. Tanulás neurális hálókkal

A neurális hálók potenciális előnyei túlmutatnak a párhuzamosságukból adódó - elsősorban hardver megvalósítás esetén mutatkozó - számítási sebességen, robusztusságon és hibatűrésen. Itt tanulási képességüket emeljük ki, vagyis azt a tulajdonságukat, hogy a beérkező minták hatására képesek struktúrájukat és/vagy súlyaikat úgy változtatni, hogy hasonló mintákra következő alkalommal már megfelelőbb választ adnak.



Annak alapján, hogy milyen információk állnak rendelkezésünkre a tanulás során - hasonlóan más gépi tanulási módszerekhez - itt is három fő tanulási elvet különböztethetünk meg.

- *Supervised learning* (felügyelt tanulás, tanulás minták alapján, tanító melletti tanulás). Ekkor a tanulóminták mellett egy, a működéssel kapcsolatos hibákat szolgáltató tanító is rendelkezésre áll. Legegyszerűbb esetben a tanítóminták mellett adottak a háló kimenetén elvárt értékek is, és ebből számíthatjuk a háló aktuális hibáját. *Reinforcement learning* (tanulás kritika mellett) esetén a megkívánt pontos kimenetek helyett csak a működés helyességét jelző információ áll rendelkezésre.
- *Unsupervised learning* (felügyelet nélküli tanulás, tanító nélküli tanulás, önszerveződés). Ekkor csak bemeneti minták állnak rendelkezésre, a helyes működésre vonatkozó bármiféle információ nélkül.
- *Self-supervised learning* (önfelügyelő tanulás). Az e kategóriába eső eljárások belső felügyeletet és hibakorrekciót alkalmaznak külső tanító nélkül.

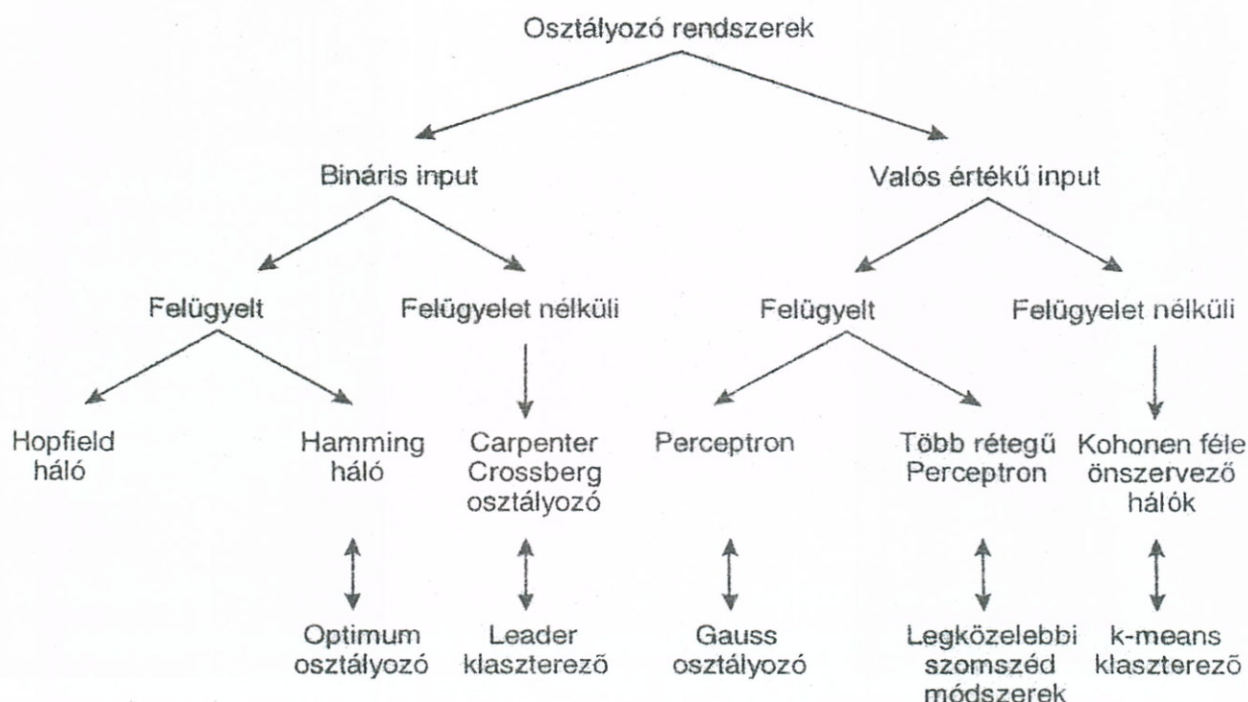
A 8. ábrán a legfontosabb neurális modellek osztályozása látható a bemenet és a felhasznált tanulási elv szerint. Az ábra bemutatja az egyes modellek a hagyományos alakfelismerési eljárásokkal való kapcsolatát is.

Neurális tanulási eljárásokat elemezve 1949-ig, a Hebb-féle tanulásig kell visszamennünk. Eszerint az  $i$ -edik és  $j$ -edik neuron közti  $w_{i,j}$  szinaptikus súly változása egyenesen arányos az érintett neuronok aktivitásával (az  $o_i$  és  $o_j$  kimenetekkel) (1).

$$\Delta w_{i,j} = k * o_i * o_j \quad (1)$$

Születése óta a fenti tanulási elvnek sok változata jelent meg, de abban közősek, hogy a súlyok változása az érintett aktivációktól és a súly korábbi értékétől függ.





8. ábra Neurális hálók osztályozása a bemenetek típusa és a felhasznált tanulási elv alapján

A neurális hálókra Rosenblatt által bevezetett, már érintett hiba visszacsatolási eljárás egy adott súly változtatásánál lokális információk helyett globális értékekből indul ki. A legegyszerűbb, egy kimenetes esetben egy  $p$  mintára a globális érték a megkívánt  $t_p$  és az aktuális  $o_p$  kimenet közti különbség:

$$E_p = t_p - o_p \quad (2)$$

Legtöbbször a mintára vonatkozó négyzetes hibát számoljuk, figyelembe véve az összes kimenetet ( $k$  a kimeneti neuronokra vonatkozó index), és ezt a hibát kívánjuk - például a legkisebb négyzetek módszerével - minimalizálni (3).

$$E_p = \frac{1}{2} \sum_k (t_{p,k} - o_{p,k})^2 \quad (3)$$

### 6.3.3. Back propagation

A legnépszerűbb, négyzetes hibaminimalizáláson alapuló tanulóeljárás több-rétegű, nemlinearitásokat tartalmazó hálókra a már említett *back propagation* eljárás, mely legegyszerűbb formájában a súlyokat az  $E$  hibafüggvény adott súlyra vonatkozó parciális deriváltjával arányosan változtatja. A  $p$  indexet elhagyva:

$$\Delta w_{j,i} = -\eta \frac{\partial E}{\partial w_{j,i}}. \quad (4)$$

Itt  $j$  valamely réteg egy neuronját jelzi,  $i$  pedig a megelőző réteg egyik neuronjára utal,  $w_{j,i}$  pedig a közöttük lévő kapcsolat erőssége (súly). Hasonlóan kezelhető a neuront jellemző küszöbérték is. Az  $\eta$  *tanulási ráta* általában konstans, és  $0 < \eta < 1$ .

A *back propagation* eljárás lényege, hogy a fenti parciális deriváltak az összetett függvények deriválására vonatkozó láncszabály alkalmazásával a háló kimeneti rétegétől indulva, rétegről-rétegre számíthatók, tehát a háló "belső" súlyaira is.

Egy adott bemeneti minta - célvektor párra az eljárás két lépcsőből áll. Az első, előre terjesztő lépésnél rétegről-rétegre kiszámítjuk a neuronok aktiválási értékeit (kimeneteit). Az ezt követő, hátra terjesztő lépésnél pedig a kimeneti rétegtől kezdve - szintén rétegről-rétegre - a súlyváltozások meghatározásához szükséges parciális deriváltakat határozzuk meg.

A *back propagation* eljárás konvergenciája nagyon lassú is lehet, sőt a perceptron tanulási eljárással ellentétben nem is mindig konvergál, gradienskövető optimálási algoritmusként lokális minimumban képes ragadni. A lokális minimum problémájára megoldást jelenthet a tanulási probléma újraindítása a súlyok megváltoztatott kezdeti értékei mellett. A konvergencia gyorsítására a legkülönbözőbb technikák ismeretesek, melyek közül talán az egyes súlyokhoz rendelt dedikált tanulási ráták módosítása a legígéretesebb.

A (4) képletet szinte kötelező az  $\alpha$  *momentummal* kiegészíteni, ahol általában  $0 \leq \alpha < 1$ , és  $n$  mutatja, hogy eddig hány - nem feltétlen különböző - mintát mutattunk be a hálónak:

$$\Delta w_{j,i}(n+1) = -\eta \frac{\partial E}{\partial w_{j,i}} + \alpha \Delta w_{j,i}(n), \quad (5)$$

A momentum-tényező egyrészt megnöveli a tanulási rátát a súlytér laposabb régióiban, másrészt csökkenti az esetleges oszcillációk bekövetkezési esélyét.

A fentiekben a súlyváltoztatásokat mintánként végeztük (*pattern learning*, minta tanulás). Szokás a változtatásokat csak akkor elvégezni, amikor már mindegyik mintát megmutattuk a hálónak (*batch learning*, kötegelt tanulás):



$$\Delta w_{j,i}(m+1) = -\eta \sum_P \frac{\partial E_P}{\partial w_{j,i}} + \alpha \Delta w_{j,i}(m). \quad (6)$$

Itt  $m$  jelöli, hogy hányszor mutattuk be már az összes mintát.

Alapvető kérdés, milyen legyen a háló struktúrája. Általában elég 3 réteget használunk, mivel a bemeneti és kimeneti neuronok száma pedig általában a feladatból következik, fő feladat tehát a középső, rejtett réteg elemszáma. Osztályozási problémák esetén létezik egy elméleti minimum a minimális elemszámra. A struktúra adaptív változtatására különböző adaptív technikák kerültek előtérbe.

#### *Kompetitív tanulás*

A kompetitív (versenyeztető) tanulás elve elsősorban a felügyelet nélküli tanuló-algoritmusokban kap helyet. Lényege, hogy a minták bemutatása során egy sereg neuron versenyez egymással, abban a tekintetben, hogy melyik kerül jobban aktív állapotba, vagyis melyiknek nagyobb a kimenete. A legegyszerűbb változatban kizárólag a győztes neuron súlyai kerülnek megváltoztatásra, oly módon, hogy ugyanazon minta újbóli bemutatásakor fölénye még nagyobb legyen (*winner takes all*, mindent a győztes visz eljárás). Eredményképp a háló az input mintákat különálló csoportokra képes bontani, mely csoportokon belül hasonló minták szerepelnek. Az érintett neurális modellek közül az ART hálók tanulási eljárásai tartalmazznak kompetitív elemeket.

#### **6.3.4. Felhasználási területek**

A mesterséges neurális hálók kutatásának 1986-tól számítható fellendülésére és újabb eredményeire gyorsan reagálva, többségében sikerrel járó kezdeményezések történtek az új technika felhasználására az élet szinte minden területén, így a technológia menedzsmenten belül is. A kutatások és az elsősorban kísérleti alkalmazások elsőrendű célja azoknak az alterületeknek a körülhatárolása volt, ahol a mesterséges neurális hálók kedvező tulajdonságaik révén hozzájárulhatnak a rendszer reakcióidejének csökkentéséhez, a termékminőség javításához, a rendszer intelligenciaszintjének és megbízhatóságának növeléséhez. A felhasználások két fő részre bonthatók:

- *általános információfeldolgozási alkalmazások*, ahol a neurális háló egy tanulási képességgel rendelkező információfeldolgozó blokként szerepel,



- *(multi)szenzor integráció*, ahol ahol a mesterséges neurális hálók alkalmazása - általában több - mért jel párhuzamos feldolgozására irányul.

Megállapítható, hogy napjaink neurális modelljei több részprobléma megoldásához járulhatnak hozzá, elsősorban azokhoz, ahol *a szenzor integráció, a magas feldolgozási sebesség, a bizonytalanság kezelésére és az adaptivitás* követelménye merül fel. Megállapítható, hogy az intelligens gyártáson belül a tradicionális, szimbolikus mesterséges intelligencia módszereknek és a mesterséges neurális hálóknak különböző de egymással átlapolódó felhasználási területei vannak. Következésképpen, intelligens rendszerek kialakítása során mindkét megközelítésre támaszkodnunk kell, olyan *hibrid AI rendszerek* kialakításával és felhasználásával, melyek egyesítik a két irányzat előnyeit.

### 6.3.5. Hibrid megközelítések

*Szimbolikus rendszerek* a tudást szabályokkal, keretekkel (frame-ekkel) vagy objektumokkal ábrázolják (2. táblázat). E tudásábrázolási formák általában áttekinthetők a felhasználók számára. A tudásgyűjtés azonban egy összetett feladatnak bizonyul, amely szakértők és tudásmérnökök összehangolt munkáját igényli. Az így kialakult szakértői rendszerek tudásbázisa viszonylag egyszerűen módosítható, pl. szabályok hozzáadásával, törlésével vagy módosításával.

*A mesterséges neurális hálók.*- mint szubszimbolikus rendszerek - a tudást implicit, az ember számára áttekinthetetlen módon, súlyok és küszöbértékek formájában tárolják. Tanulás folyamán struktúrájuk dinamikusan változik. Néhány ANN modellnél önszervező mechanizmusok léteznek, melyek lehetővé teszik a felügyelet nélküli tudásgyűjtést. A hálók valamely komponensének megváltoztatása több modellnél lényegében a teljes tanulási folyamat megismétlését jelenti (2. táblázat).

Következésképpen igen nagy jelentőségük van azoknak a törekvéseknek, melyek a két módszer előnyeit ötvözve, hibrid rendszerek létrehozását célozzák. A megoldások a következő csoportokba oszthatók (9. ábra):

- *Különálló (stand-alone) modellek*, amikor a szimbolikus és szubszimbolikus alrendszerek egymással nem kommunikálnak, de a felhasználó mindkettőre támaszkodhat.
- *Transzformációs (transformational) modellek*. A szubszimbolikus-szimbolikus irányban a neurális rész képes adaptálódni összetett, adatintenzív problémákhoz, általánosításokat hozni és hibás adatokat

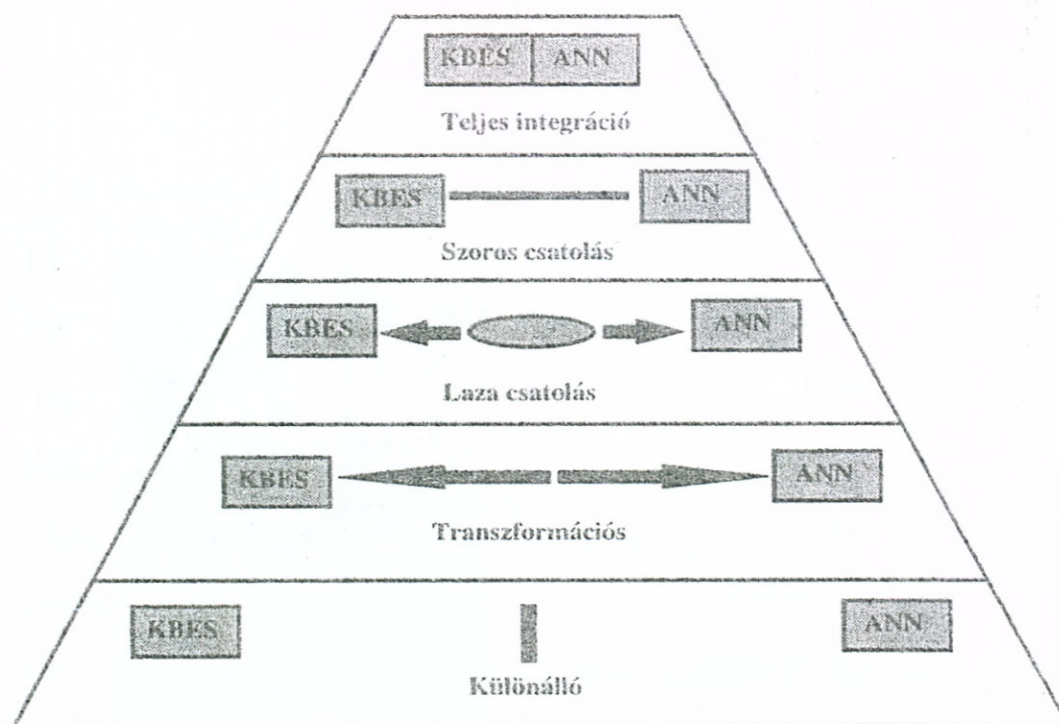


kiszurni. A kevésbé gyakori szakérto rendszer (KBES)- neurális háló (ANN) irányban a szakérto rendszerben tárolt tudást alkalmazzák a hálók inicializálására és tanítására.

- *Lazán csatolt (loosely-coupled)* modellekben a két komponens adatfile-okon keresztül kommunikál egymással.
- *Szorosan csatolt (tight coupled)* modellek memóriarezidens adatstruktúrákon tartanak egymással kapcsolatot.
- *Teljesen integrált (fully-integrated) hibrid modellek* komponensei közös adatstruktúrákkal és tudásábrázolási technikával rendelkeznek. Ide tartoznak a napjainkban egyre gyakoribb neuro-fuzzy rendszerek.

Tudás	Szimbolikus rendszerek	Neurális hálók
Ábrázolás	<b>Explicit</b> - szabályok - keretek <b>Áttekinthető</b>	<b>Implicit</b> - súlyok - küszöbértékek <b>Áttekinthetetlen</b>
Gyűjtés	<b>Nehéz</b> - tudásmérnökök - szakértők	<b>Könnyű</b> - önszervezés
Módosítás	<b>Könnyű</b> - hozzáadás - törlés - módosítás	<b>Többnyire nehéz</b> - tanulás újratekdése

2. Táblázat Szimbolikus rendszerek és mesterséges neurális hálók összehasonlítása



9. ábra Szimbolikus és szubszimbolikus rendszerek integrálási módjai

## ELŐADÁS

A mesterséges intelligencia alapjai

### Nem-szimbólikus módszerek

#### Neurális hálók

Az emberi idegrendszer, agy struktúrájának, működésének vizsgálata vezetett el ide.

A természetes neuronok fizikai modellezése mesterséges (elektronikai) eszközökkel.

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Az emberi agy felépítése és működése

Az agy és az emberi idegrendszer alapvető egysége a neuron

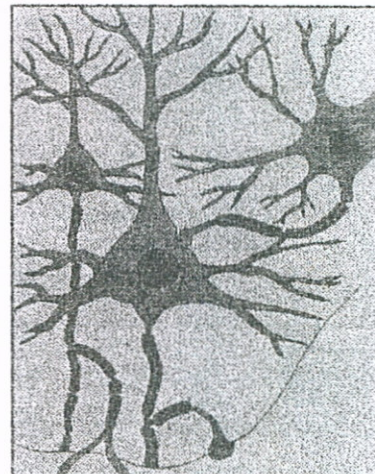
sejttest (szoma)

sejtmag

axon

dendritek

szinapszis



Dr. Szalay Tibor



A mesterséges intelligencia alapja:

## Mesterséges neurális hálók

A mesterséges neurális hálók (artificial neural networks, ANNs) "egyszerű, általában adaptív elemek sűrűn összekapcsolt hálózatai, és ezek hierarchikus szervezetei, amelyek a valós világ objektumaival olyan kölcsönhatásban állnak, mint a biológiai idegi rendszerek".



Dr. Szalay Tibor

!

A mesterséges intelligencia alapja:

## Történeti áttekintés

1943 McCulloch és Pitts

1958 Rosenblatt

1960 Widrow és Hoff

1962 Rosenblatt *többszintű perceptron*

1969 Minsky és Papert

1982 Hopfield

1986 Rummelhat, Hinton és Williams

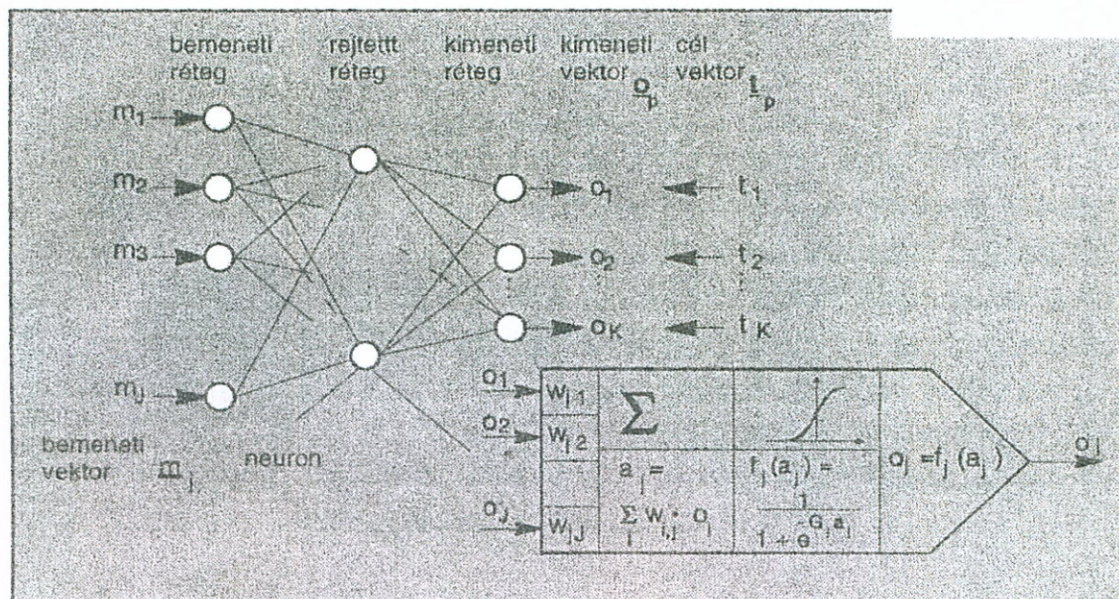
1989 Kohonen

1991 Carpenter és Grossberg adaptive resonance theory ART

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

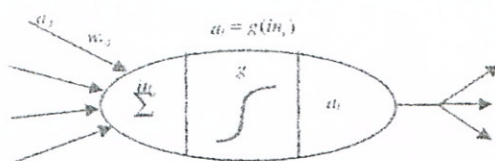
### Mesterséges neurális hálók felépítése



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Alapegység



- bemenet, aktiválási függvény, kimenet
- küszöbérték, átviteli függvény
- számítás két lépésben
  - bemenet: lineáris összegzés
  - kimenet: nemlineáris
- kimenet: 1/tüzelés, 0/nincs tüzelés

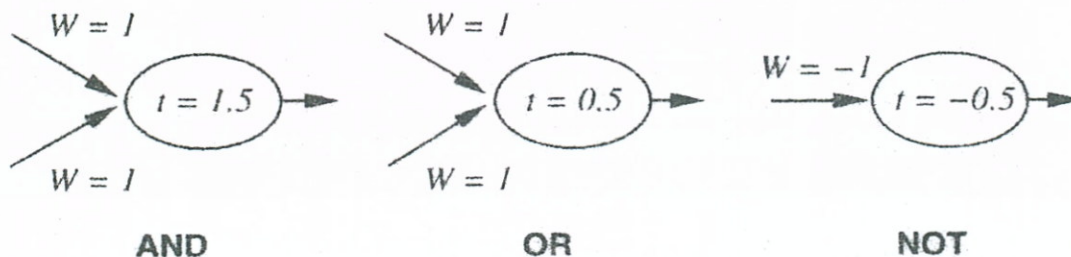
Dr. Szalay Tibor



A mesterséges intelligencia alapjai

### Percepton

- egy bementi és egy kimeneti réteg
- előrecsatolt
- egyszerű példák: logikai függvények



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

### Tulajdonságok

A háló topológiája  
 A neuronok karakterisztikája  
 Az alkalmazott tanulási algoritmus

} határozza meg

### Topológia

előrecsatolt	teljesen csatolt
visszacsatolt	ritkán csatolt

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**előrecsatoló (feed-forward)**

egyirányú élek (ciklusmentes)

kimenetet számol a bemenete függvényeként,  
ez csak a súlyoktól függ

nincs más belső állapot, csak a súlyok

tanulás - nemlineáris regresszió

**visszacsatoló (recurrent)** pl. Hopfield-háló

az élek tetszőleges topológiát alkothatnak

vannak visszafele mutató élek

a belső állapot az egységek aktivációs szintjeként  
van tárolva

Dr. Szalay Tibor

A mesterséges intelligencia alapjai

**Neuronok karakterisztikája**

Egy belső küszöbérték és átviteli függvény jellemzi

Példa:

Ugrásfüggvény

$$\text{step}_t(x) = \begin{cases} 1, & \text{ha } x \geq t \\ 0, & \text{ha } x < t \end{cases}$$

Előjelfüggvény

$$\text{sign}(x) = \begin{cases} +1, & \text{ha } x \geq 0 \\ -1, & \text{ha } x < 0 \end{cases}$$

Sigmoidfüggvény

$$\text{sigm}(x) = \frac{1}{1 + e^{-x}}$$



Dr. Szalay Tibor



### Tanulási algoritmus

- 1) Legyenek az  $n$  háló súlyai véletlen módon adva
- 2) Amíg minden példát meg nem tanult,
- 3) Minden  $e$  példára egyenként:
  - 3a) állítsd elő  $O$  kimenetet,
  - 3b) hasonlítsd össze a kívánt  $T$  értékekkel,
  - 3c) az eltérés függvényében módosítsd  $n$  súlyait.

Hiba

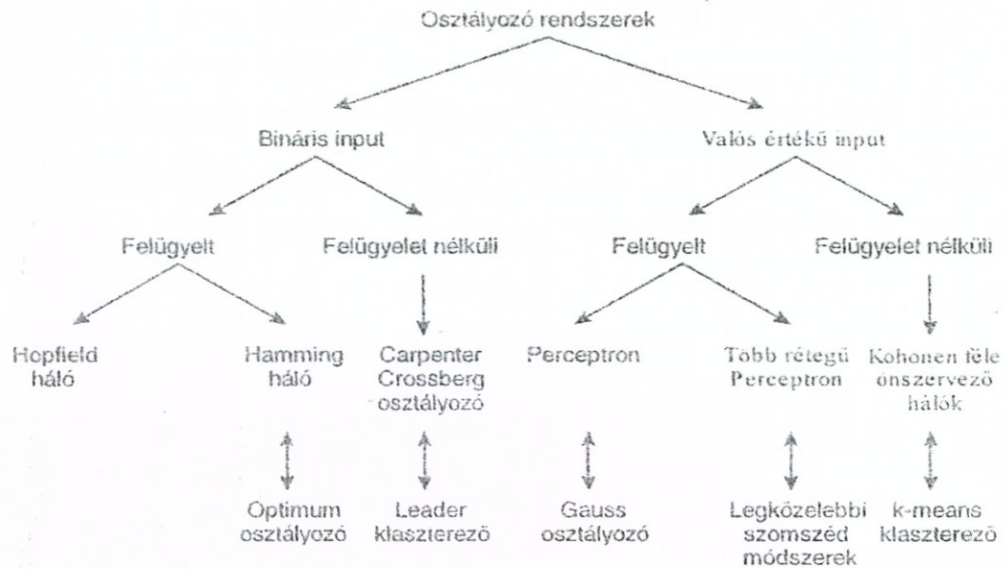
$T$  (target/cél),  $O$ =output  
halmazozott hibák négyzetösszege

$$E_p = \frac{1}{2} \sum_k (T_{pk} - O_{pk})^2$$

### Tanulás

felügyelt tanulás: tanulás minták alapján  
tanulás kritika mellett (*reinforcement learning*)  
felügyelet nélküli tanulás (önszerveződés)  
ön-felügyelő tanulás (belső korrekció)

## Neurális hálók osztályozása bemenetek típusa és tanulási elv alapján



Dr. Szalay Tibor

## Backpropagation

- többrétegű nemlineáris hálóban
- a hiba értékének függvényében a súlyokat visszafelé állítja át a hiba minimalizálása érdekében
- rétegről-rétegre, az output-tól az input felé

$$\Delta w_{j,i} = -\eta \frac{\partial E}{\partial w_{j,i}}$$

avagy, ún. momentum tényezővel, ha már  $n$  mintát látott:

$$\Delta w_{j,i}(n+1) = -\eta \frac{\partial E}{\partial w_{j,i}} + \alpha \Delta w_{j,i}(n)$$

Dr. Szalay Tibor



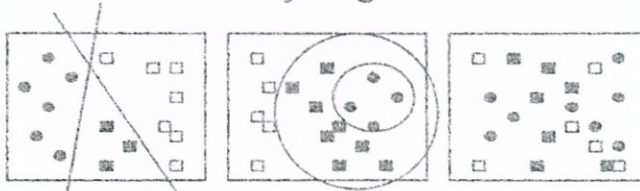
### 6.3.6. Alkalmazás

A mesterséges intelligencia típusai

#### Alkalmazás

##### Feladattípusok

- osztályozás
- az információmennyiség erős csökkentése



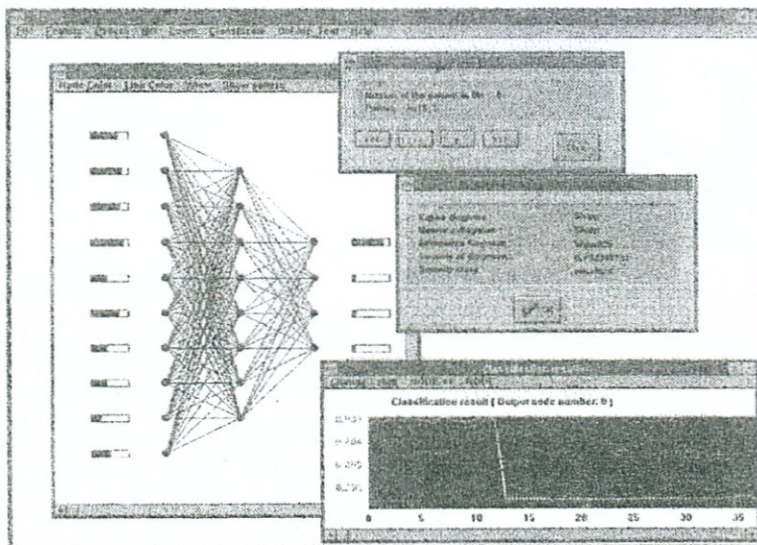
- transzformáció  
egyik adatstruktúrából egy másikba való átalakítás  
pl. írott szöveg ! beszéd
- optimalizálás

Dr. Szalay Tibor

A mesterséges intelligencia típusai

#### Példa

### Szerszámkopás felügyelet



Dr. Szalay Tibor





## 7. FIZIKAI ÁGENSEK – ROBOTIKA

Az ipari robot szabadon programozható, többcélú manipulátor, mely anyag, alkatrész, szerszám, vagy egyéb eszköz egyszerűen változtatható program szerinti mozgását; térbeli helyzetének megváltoztatását vagy megtartását, megfogását vagy elengedését, vagyis manipulálását végzi.

Az ipari robotok kinematikai felépítése sokféle lehet. Ez jelentősen befolyásolja az ipari robot munkaterének alakját, terhelhetőségét, működési sebességét és különféle pontos-sági paramétereit. Az ipari robotok alkalmazástechnikai vizsgálatát nemzetközi szabvány is rögzíti (Manipulating industrial robots- Performance criteria and related test methods ISO 9283). Az ipari robot kinematikai felépítését (karrendszer) a szabadságfokok és így a translációs (T) vagy Rotációs (R) mozgástengelyek száma, rendszerei meghatározzák. (A megfogószerkezet nyitás/zárás funkcióját nem számítjuk szabadságfoknak). Ahhoz, hogy az ipari robot effektorát (megfogó, szerszám) a munkatér meghatározott helyére és meghatározott helyzetbe hozzuk legalább  $3+3+6$  szabadságfokú karrendszerre van szükség. A robotalkalmazások többségéhez ennél kevesebb szabadságfok is elegendő. Az ipari robotokat általában az első három csukló elrendezése alapján legtöbbször a következő öt csoportba szokás sorolni. (10. ábra).

(A legtöbb országban a három szabadságfokunál egyszerűbb karrendszert nem tekintik robotnak).

### **Hasáb munkaterű ipari robot**

Csak lineáris mozgástengelyekkel rendelkezik. Vezérlése egyszerű, hasáb alakú munkatere méretéhez viszonyítva kicsi, viszont nincs ún. holt tere. Portál változata forgácsoló szerszámgépek kiszolgálására előszeretettel használatos. Az ipari robotok mintegy 40%-a ilyen.

### **Henger(koordinátás) munkaterű robot**

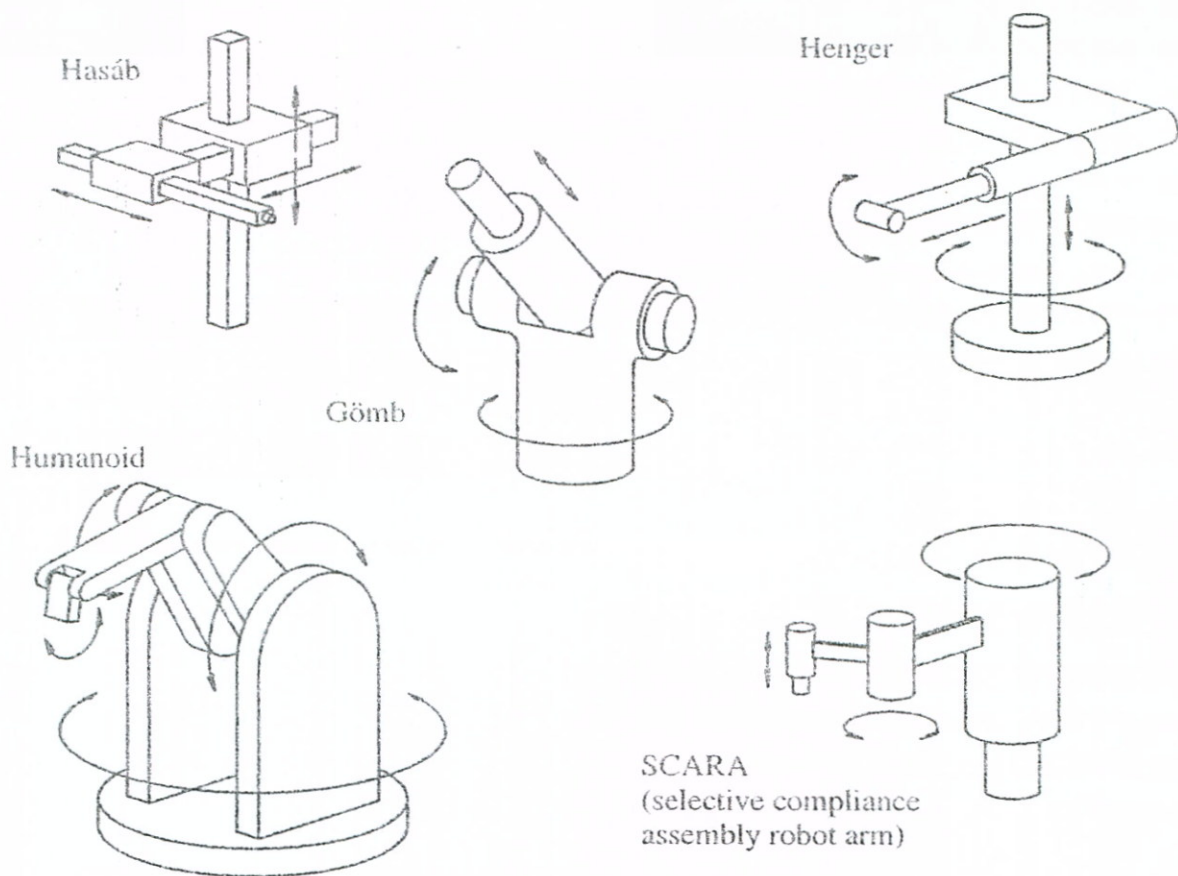
Ez a robottípus mely egy rotációs és két translációs mozgástengellyel rendelkezik üreges henger munkaterű. Forgácsoló szerszámgépek kiszolgálására ez is alkalmas. Átmenetet képez a hasáb munkaterű és az ún. SCARA robot között.

## Gömb(koordinátás) munkaterű ipari robot

Egy translációs és két rotációs mozgástengely/szabadságfok jellemzi. Munkatér variációi a felépítéstől függően gömbszegmenstől teljes (üreges) gömb munkatérig többfélék lehetnek. Gépkihasználásra ez is alkalmas, de viszonylag kis munkatere miatt nem igazán kedvelt.

## Humanoid robot

Jellemzője, hogy kizárólag rotációs szabadságfokú elemekből áll. A legelterjedtebb változata 6 szabadságfokú. Ez a robottípus univerzálisan alkalmazható, munkatere méretéhez viszonyítva nagy. A vezérléssel szemben támasztott követelmény nagy. (L.transzformációk) Jól bevált robottípus. Az ipari robotok kb. 40% ilyen.



10 ábra. Ipari robotok alaptípusai

## SCARA robot

A Selective Compliance Assembly Robot Arm (szelektív engedékenyséű robot kar) három rotációs és egy translációs tengelyből áll. Mint elnevezése is mutatja szerelési, behelyezési feladatokra (pl. nyomtatott áramköri lapba elemek beillesztése) fejlesztették ki. Nagy működési sebesség és nagy is-



méltési pontosság jellemzi. Terhelhetősége ugyanakkor általában kicsi. Annak ellenére, hogy kb. tíz éve jelent meg, a robotpiac 10%-át már elfoglalta.

## 7.1. Az ipari robotok mozgástörvényei. A robotkarok kinematikája

Ha az effektor előírt háromdimenziós pályája, a pályabejárás mozgástörvénye, mint időfüggvény a derékszögű koordinátarendszer abszolút koordinátaiban – világkoordináták – van meghatározva, akkor a pálya megvalósításához a robotmanipulátor koordinátái ( $###, S_i$ ) (L.11.ábrát!) időfüggvényeinek a meghatározása szükséges ( $i###1, 2, \dots, n$ ). Ez a programozott pont (TCP) derékszögű világ-koordinátáiból a robotspecifikus karkoordinátákba történő koordináta-transzformációt igénylő feladat. Amennyiben a karkoordinátákat, csuklósögeket ismerjük (például szenzoradatokból), akkor az effektor helyzetének és irányának meghatározásához egy, a robot-koordinátákból a világ-koordinátákba történő koordináta-transzformáció szükséges. A korszerű robotvezérlők a mozgáspálya tervezéshez, vezérléshez mindkét transzformációt el tudják végezni. A két transzformáció röviden a következőképpen jellemezhető:

A kar (robot) koordináták világ-koordinátákba transzformáció a Denavit-Hartenberg-transzformációs mátrixszal (továbbiakban D-H mátrix) történik és mátrixok szorzásával egyértelmű megfeleltetést jelent.

A világkoordináták visszatranszformálása robotkoordinátákba nem egyértelműen meghatározott. A számítás nagymértékben függ a kar geometriájától és gyakran több megoldást eredményez. (Ez különösen az un. redundáns kinematikai láncoknál okoz nehézséget.) A robotkoordináták világkoordinátákba transzformálásához a kar D-H szerinti mátrixos leírása szükséges, melynél a kar minden mozgástengelyére egy derékszögű koordinátarendszert ültetünk olyan irányszabályok szerint, hogy két egymást követő koordinátarendszer esetén:

$###$  a  $z_{i-1}$  tengely az  $i$ -dik karelem mozgástengelye irányában fekszik;

$###$  az  $x_i$  tengely  $z_{i-1}$  és  $z_i$  normál transzverzálisa

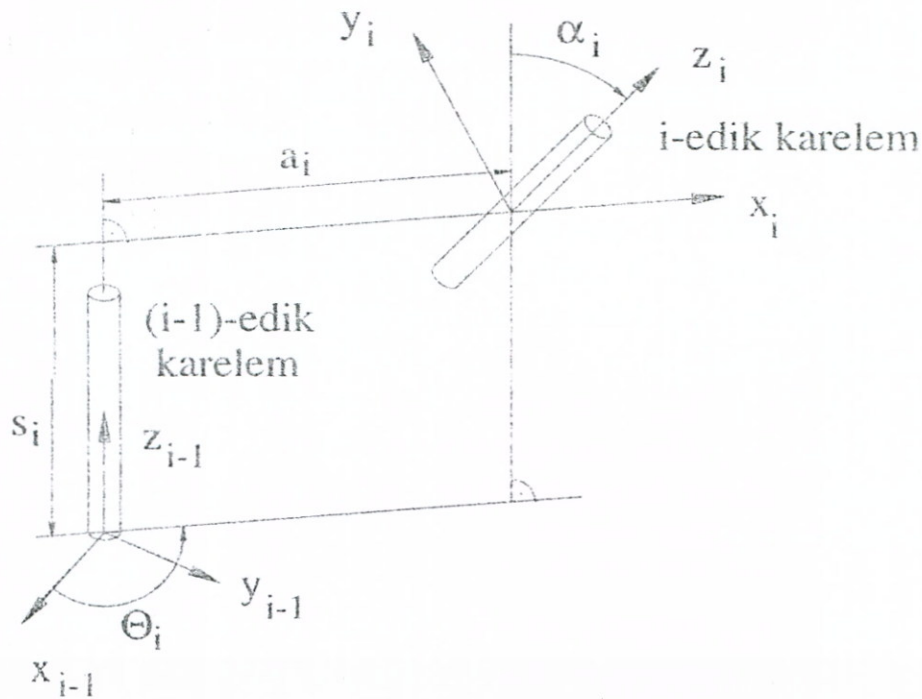
Az ily módon meghatározott koordinátarendszerek két egyenesvonalú és két forgó mozgással egymásba átvihetők. A 11. ábrán:

- ### ###, elfordulás  $z_{i-1}$  körül:  $x_{i-1}$  párhuzamos lesz  $x_i$ -vel,
- ###  $S_i$  elmozdulás  $z_{i-1}$  mentén  $z_{i-1}$  és  $x_i$  metszéspontjáig,
- ###  $a_i$  elmozdulás  $x_i$  mentén: a koordinátarendszerek kezdőpontja fedésbe kerül,
- ### ###, elfordulás  $x_i$  körül: a  $z$  tengelyek fedésbe kerültek.

Fenti négy mozzanat a 4###4-es D-H transzformációs mátrixban van összefoglalva:

$$D_{i-1,i} = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \cdot \sin \theta_i & -\sin \alpha_i \cos \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \sin \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & s_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A D-H mátrix egy pontot homogén koordinátákban átszarmaztat az  $i$ -edik robot koordinátarendszerből az  $i-1$ -edikbe



11. ábra. A szomszédos derékszögű koordináták helyzete Denavit-Hartenberg szerint



$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = D_{i-1,i} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

szerint.

A robotmanipulátor kar összes  $D$ - $H$  mátrixának összeszorzásával:

$$E = D_{0,1} \cdot D_{1,2} \cdot \dots \cdot D_{n-1,n}$$

ismét egy  $4 \times 4$ -es mátrixot kapunk, mely az effektor helyét és irányát világkoordinátákban adja meg:

$$E = \begin{bmatrix} x_{ex} & y_{ex} & z_{ex} & x_s \\ x_{ey} & y_{ey} & z_{ey} & y_s \\ x_{ez} & y_{ez} & z_{ez} & z_s \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ha a csuklózögeket ismerjük az  $E$  mátrix segítségével az effektorhoz rögzített koordináta-rendszer origójához húzott  $F$  helyvektor (koordinátái  $x_s, y_s, z_s$ ), és a koordináta-rendszer egységvektorai ( $\bar{x}_e, \bar{y}_e, \bar{z}_e$ ) meghatározhatók.

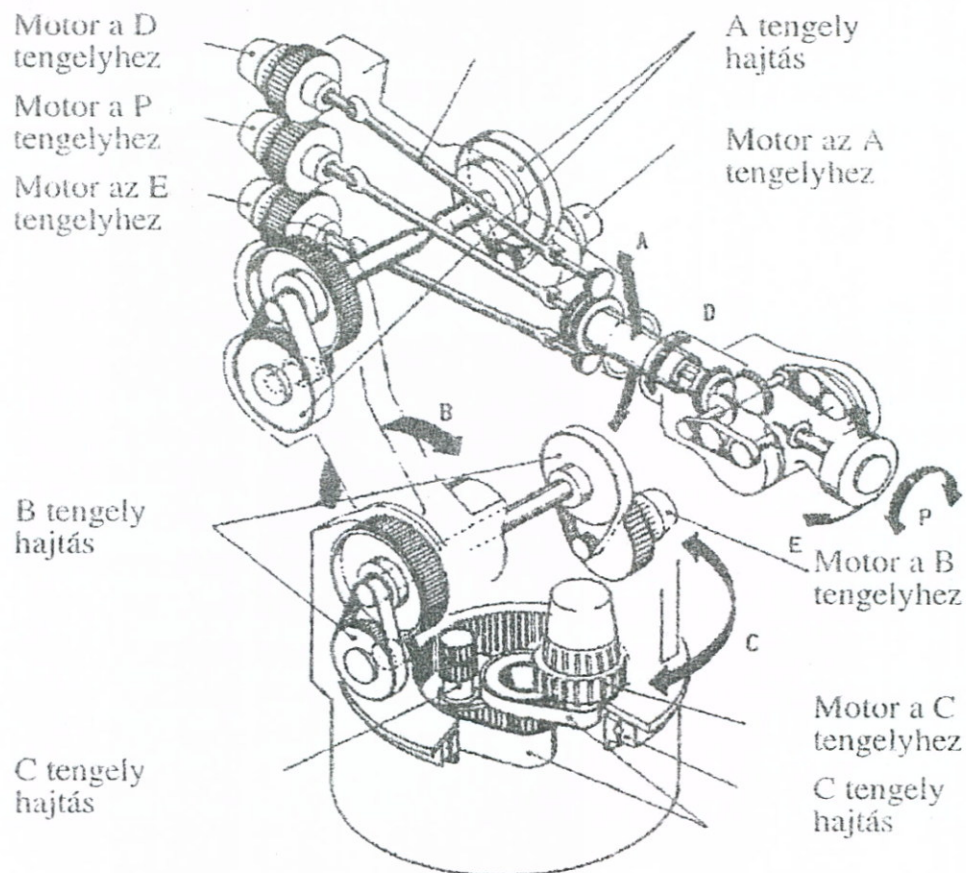
## 7.2. Jellegzetes ipari robot építőelemek

Az ipari robotok lényegében a forgácsoló szerszámgépekével azonos, vagy nagyon hasonló részegységekből állnak. A különbségeket a súlycsökkentés iránti fokozottabb igény, a lényegesen kisebb merevségi, pontossági követelmények, a nyitott karrendszerből eredő konstrukciós feladatok (egymásba csapágyazott csuklók stb) okozzák. Az elektronikus hajtások térhódítása a robotoknál is szembevetendő. Az ipari robotok vázszerkezete, karelemei vékonyfalú alumínium vagy acél öntvényből, hegesztett lemezszerkezetből, esetleg szénszál-erősítésű műanyagból készülnek. Fő szempont a nagy merevség/súly arány. Vezetékeknél, csapágyazásoknál a legfontosabb eltérés: speciális, növelt belső átmérőjű, az egymásba vezetést biztosító gördülőcsapágyak.

A robotok minden egyes szabadságfokához külön hajtáslánc tartozik. Ennek elemei: hajtómotor/tápegység, hajtómű, mozgásátalakító, vezérlő/szabályozó egység. A megfogószerkezeteknek általában szintén külön hajtásláncuk van. Az energiaellátás szempontjából megkülönböztethető villamos, hidraulikus és pneumatikus, esetleg ezek kombinációjából álló ipari robot hajtáslánc rendszer. Ezek mindegyikének több, robottechnikai szempontból fontos előnye illetve hátránya van. A következőkben ezeket foglaljuk össze:

### Villamos hajtás

Az ipari robotoknál a kevés karbantartást igénylő váltóáramú (AC) villamos szervóhajtások a legelterjedtebbek. Előnye hogy a villamos energia könnyen hozzáférhető; a fokozat nélküli hajtás gyakorlatilag holtidőmentes; egyszerű, kevés karbantartást igénylő megoldás; zajmentes; megbízható. Hátránya hogy közepes nyomatékú; nagy fordulatszámú; robbanásbiztos megoldása (ha igény van rá) bonyolult és költséges; az ún. direkthajtású kivitel csak kis terheléseknél, nagy sebességű robotoknál lehetséges, nagyobb nyomatékigénynél, lassú mozgásnál redukáló hajtómű pl. a különleges mintegy 200-as lassítóáttételű hullámhajtómű alkalmazása szükséges. Költséges megoldás.



12. ábra Hat szabadságfokú csuklós robot (KUKA)



A 12. ábrán egy hat szabadságfokú, 60 kg névleges terhelhetőségű csuklós robot látható (KUKA). Mind a hat mozgástengely egyenáramú (DC) tárcsamotorral hajtott melyek beépített állandó mágneses biztonsági fékekkel vannak ellátva. (A mozgás során a fékhatást természetesen megszünteti egy ellenmágnes tér). A motorok fordulatszámát fogazott műanyagszíj hajtással és fogaskerék áttételekkel, majd az ábrán nem látható hullámhajtóművekkel redukálják, ezzel a nyomatékot növelik.

A szervó motort gördülő orsó-anya párral egybeépítve jelentős billentő nyomatékú mozgást lehet megvalósítani. (nagy méretű robotoknál gyakori megoldás)

### Hidraulikus hajtás

Egyenesvonalú és forgó mozgást is lehet hidraulikus hajtóművel létesíteni. Előnye hogy nagy erőkifejtésre képes; fokozat nélküli sebességszabályozást biztosít; rögzítőféket nem igényel (l. alább); a hidraulika olaj összenyomhatatlan. Hátránya hogy hidraulikus tápegység kell; kis, hőmérsékletfüggő pozicionálási pontosság; jelentős zaj.

### Pneumatikus hajtás

A robotoknál elterjedt elektropneumatikus hajtásoknál a pneumatikus hajtóelemek, a sebesség szabályozása, pozicionálása, elektromágneses fékekkel történik. Előnye hogy préslevegő hálózat általában adott; kedvező teljesítmény/súly arány; kevés karbantartást igényel; robbanásbiztos. Hátránya hogy korlátozott erő/nyomaték; jelentős zaj; nagy fék kopás; kis pozicionálási pontosság.

Az ipari robotokba a hidraulikus működtetésűek kivételével, minden mozgástengely rögzítésére külön féket kell beépíteni. Ezek elsődleges feladata a karok, csuklók rögzítése a súlyerő ellenében. Egyúttal vészleállások esetén az azonnali, biztonságtechnikai előírásokban megkövetelt mozgásleállítást, karrögzítést is biztosítják.

Az ipari robotok nagyon fontos részei a **megfogó szerkezetek**. Ezek nélkül a robot nem tudná ellátni anyagmozgatási feladatát. Az ideális ipari robot-megfogószerkezet az emberi kéz gépi másolata lenne. Nem véletlen, hogy az emberi kéz vizsgálata a sokoldalú robot megfogó szerkezetek tervezésének egyik kiindulópontja.

A megfogó szerkezet kialakítása nagymértékben függ annak alkalmazásától. A mozgatandó tárgy alakja, mérete, anyaga, az anyagmozgatási feladat végrehajtásának körülményei, környezete annyira sokféle lehetnek, hogy szinte minden feladatra speciális megfogó szerkezetet alkalmaznak. Mivel a megfogó szerkezet alakja, mérete, feladata számtalan variációt eredményez, nincs általános – az emberi kézhez hasonló – felépítésű megfogó szerkezet, bár kialakítására több próbálkozás is történt. Az emberi kéz és a műkezek, megfogó szerkezetek közötti távolság még sokáig megmarad, talán soha nem tűnik el egészen.

A megrendelők igényeinek gyors kielégítésére a robotgyártók, újabban a megfogó-szerkezet – gyártásra szakosodott cégek, a megfogószerkezetek és azok elemei széles választékát készítik és megfogó szerkezet tervezési példatárakat, katalógusokat adnak ki. A megfogószerkezetek különféle elven működnek. Általános ipari alkalmazásnál a mechanikus, fogó típusú megfogó szerkezet a leggyakoribb. A robotok megfogószerkezetét, általánosabban az aktuátorokat felfogó felülete szabványosított (ISO 94.09), és a megfogószerkezetek (aktuátorok) automatikus cserélését lehetővé tevő mechanikus interfész tervezési elveinek nemzetközi szabványosítása is folyamatban van. (ISO/CD 11593). (A megfogószerkezet gyártására szakosodott cégek már ilyen interfészeket is gyártanak).

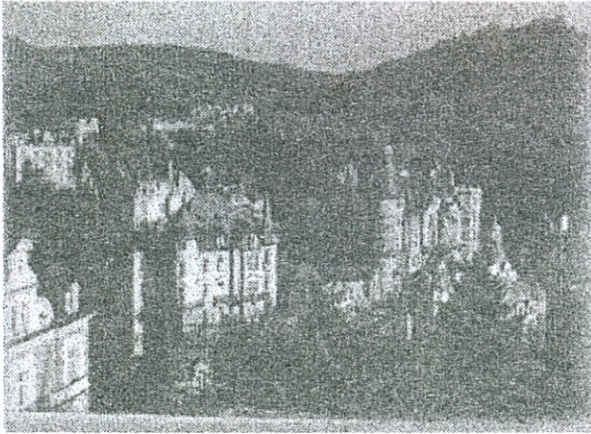
A robotmegfogószerkezetek megbízható működtetése a robot intelligenciájának fontos részét képező különféle (pl. a szorítóerőt felügyelő) érzékelők nélkül elképzelhetetlen.



# ELŐADÁS

A mesterséges intelligencia alapja

## GOLEM



A legenda a középkori Págaból származik, Loewy abbé meséjéből.

Az agyagból gyúrt Gólem az emberek védelmére és segítségére készült, egyre több emberi tulajdonságot adtak (tanítottak) neki.

Amikor felismerte, hogy hasonló az emberhez, már nem szolgálta őket.



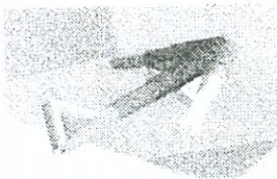
Dr. Szalay Tibor

1

A mesterséges intelligencia alapja

## GOLEM

### Genetically Organized Lifelike Electro Mechanics



A projekt elemei: Automatikus tervezés (evolúciós)

Automatikus gyártás (rapid prototyping)

A motort kívülről adták hozzá

Első lépésként értékelhető az önmagát megalkotó, javító, sokszorosító, emberi tulajdonságokat mutató eszközök terén

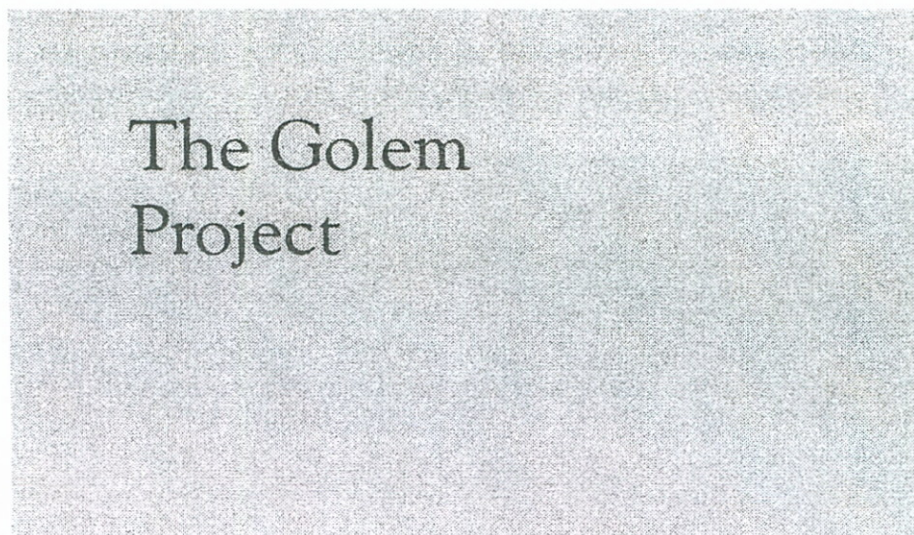


Dr. Szalay Tibor

2

A mesterséges intelligencia alapjai

## Video



Dr. Szalay Tibor

3

A mesterséges intelligencia alapjai

## Robot



Karel Capek színdarabja nyomán az  
automatikus működésű eszköz neve:

ROBOT - a szláv robota szóból ered  
(**"Rossum's universal robot"**)

Dr. Szalay Tibor

4



## Definíció

Ipari robotoknak azokat a szabadon programozható, többcélú mechanizmusokat nevezzük, amelyek anyag, alkatrész, szerszám vagy egyéb eszköz egyszerűen változtatható program szerinti mozgását, térbeli helyzetének megváltoztatását vagy megtartását, megfogását vagy elengedését, vagyis manipulálását végzik.



## A robotok alkotóegységei

### Mechanika

A tárgy pozicionálását és mozgását biztosítja

### Effektorok

A tárgy megfogását vagy megmunkálását végzi

### Motorok

A mozgás vagy az effektorok számára szükséges energiát biztosítja

### Szenzorok

Érzékeli vagy analizálja a mechanizmus aktuális állapotát illetve környezetét

### Vezérlés

A robotmechanizmus mozgását szinkronizálja

### Számítógép

A robotprogram szerkesztése és futtatása, a robot tesztelése

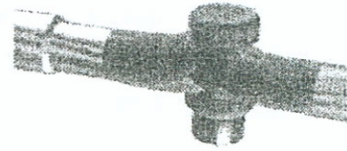
A mesterséges intelligencia alapjai

## A robotcsuklók fajtái



### Forgó (Rotációs) csuklók

- A) Hajlító (Bending)  $\psi$   $\phi$   
 B) Csavaró (Twist)  $\otimes$   $\ominus$



### Lineáris csuklók

- A) Egyenesbevezetés  $\equiv$   
 B) Teleszkóp  $\equiv$

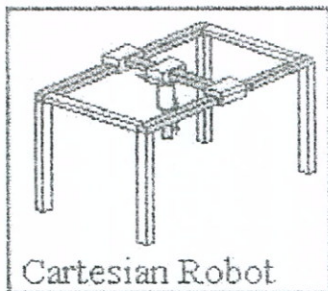


Dr. Szalay Tibor

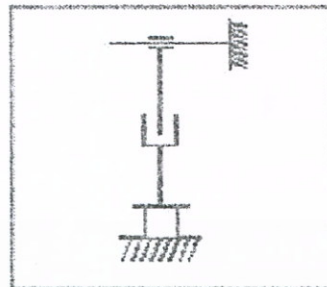
7

A mesterséges intelligencia alapjai

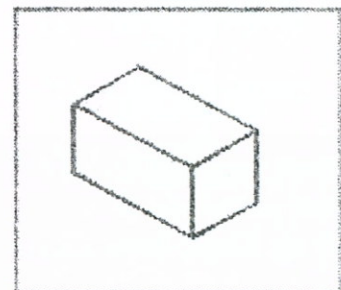
## Derékszögű koordinátás robot



Koncepció



Struktúra



Munkatér

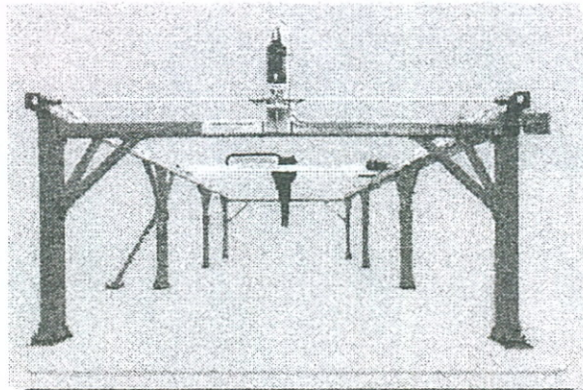
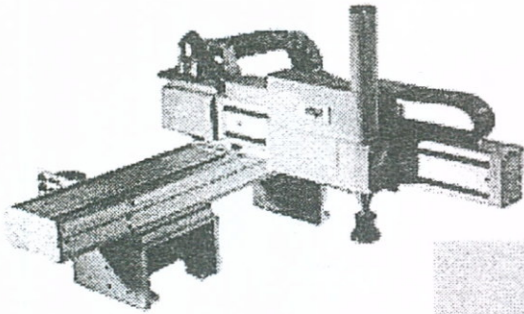
Dr. Szalay Tibor

8



A mesterséges intelligencia alapjai

## Derékszögű koordinátás robot

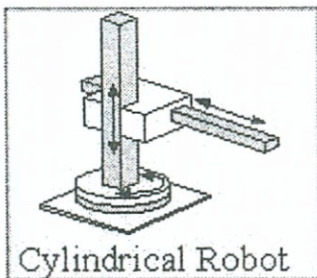


Dr. Szalay Tibor

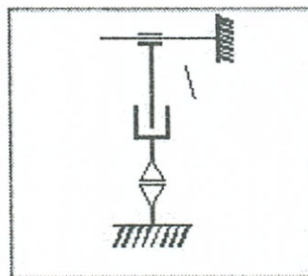
9

A mesterséges intelligencia alapjai

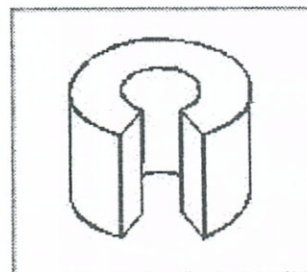
## Henger koordinátás robot



Koncepció



Struktúra

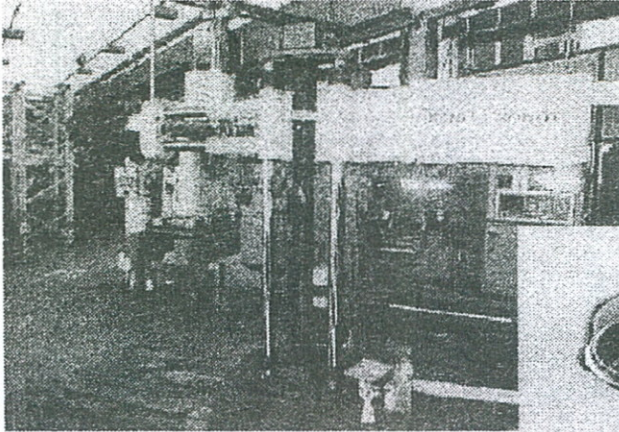


Munkatér

Dr. Szalay Tibor

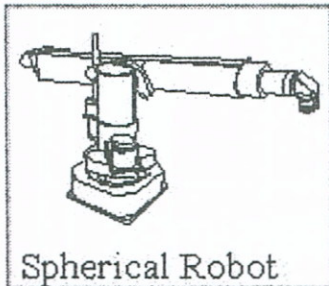
10

## Henger koordinátás robot



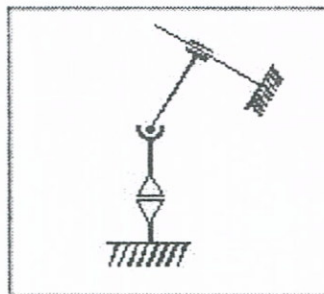
A mesterséges intelligencia alapjai

## Gömb koordinátás robot



Spherical Robot

Koncepció



Struktúra



Munkatér

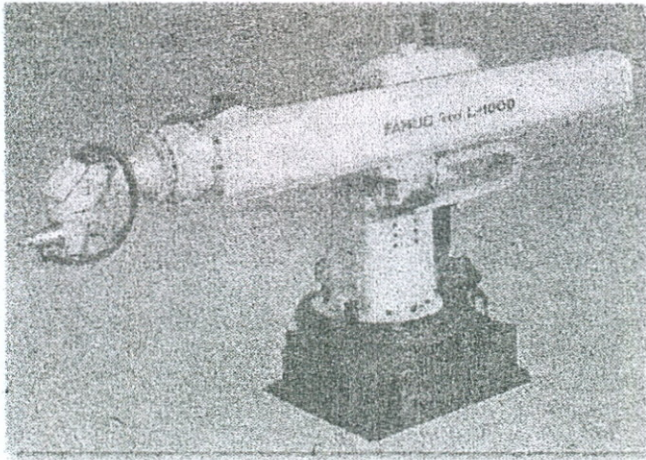
Dr. Szalay Tibor

12



A mesterséges intelligenciás alanya

## Gömb koordinátás robot

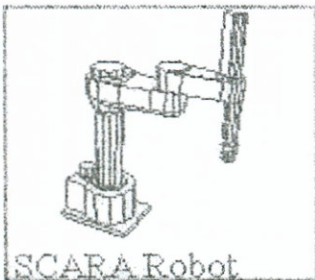


Dr. Székely Tibor

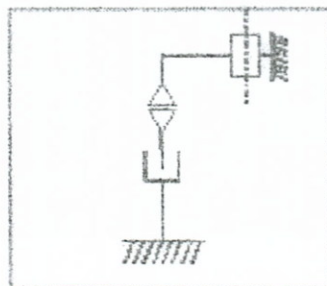
13

A mesterséges intelligenciás alanya

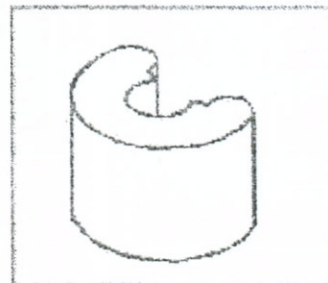
## SCARA robot



Koncepció



Struktúra



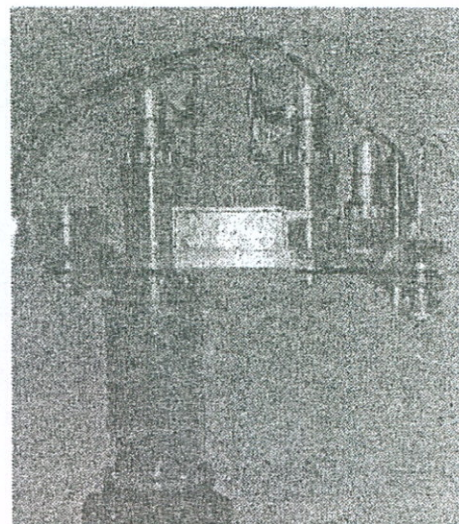
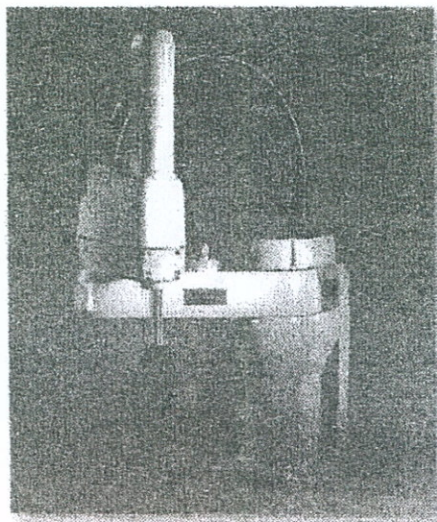
Munkatér

Dr. Székely Tibor

14

A mesterséges intelligencia alapja:

### SCARA robot

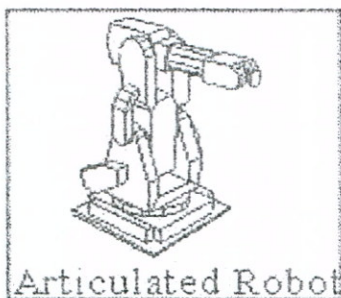


Dr. Szalay Tibor

13

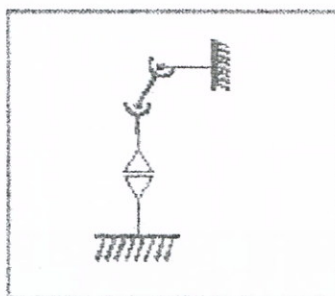
A mesterséges intelligencia alapja:

### Humanoid robot

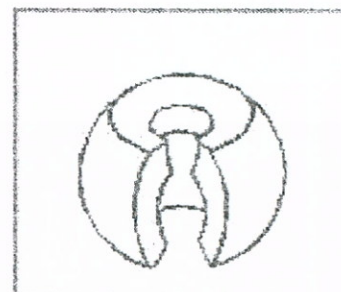


Articulated Robot

Koncepció



Struktúra

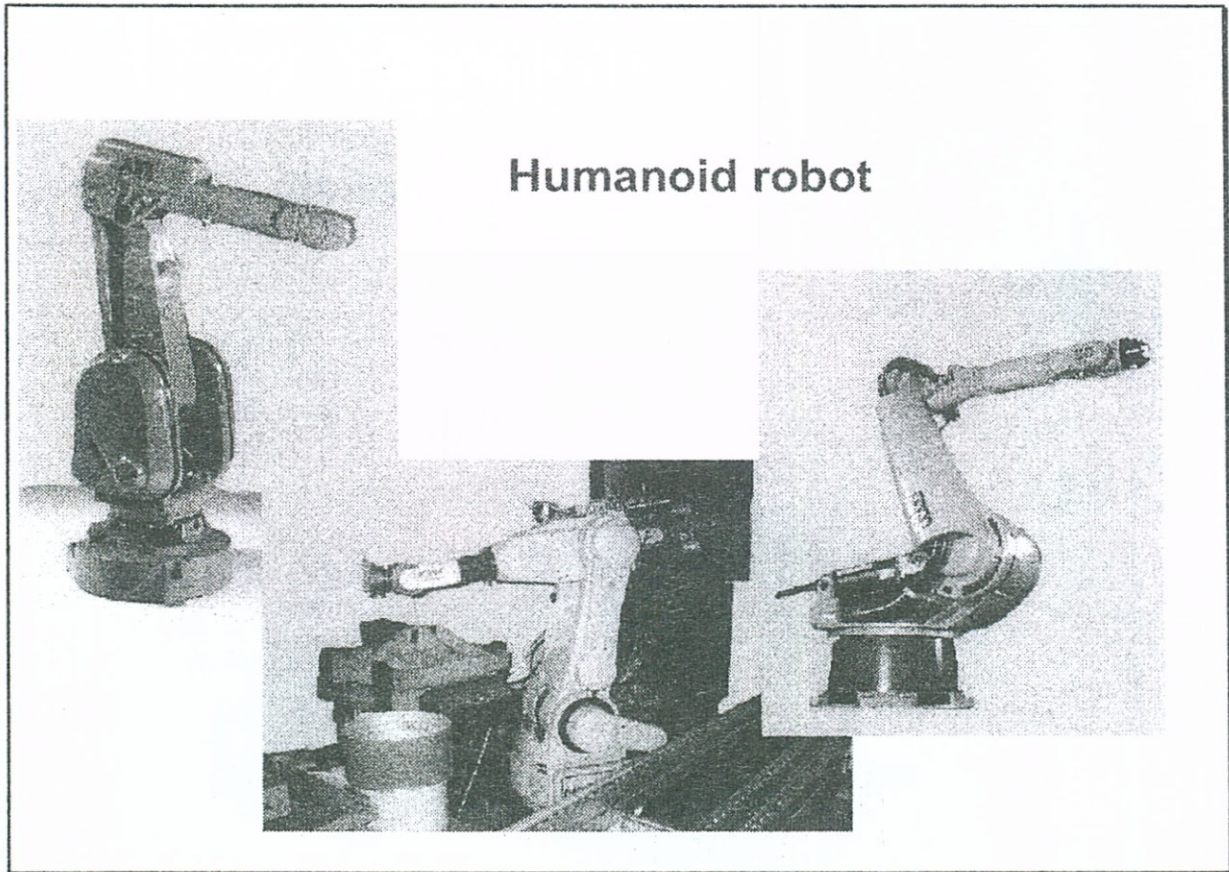


Munkatér

Dr. Szalay Tibor

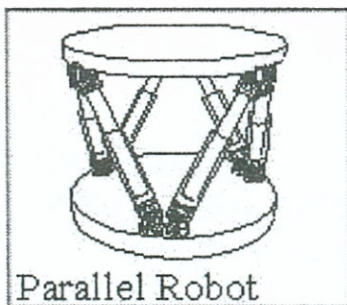
14





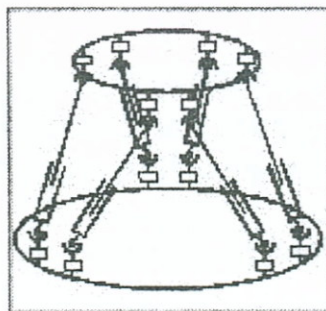
A mesterséges intelligencia alapja

### Párhuzamos kinematikájú robot

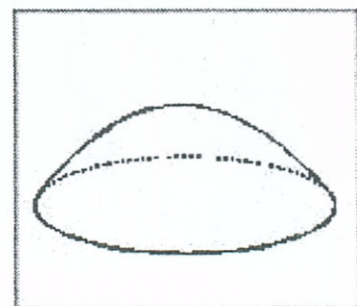


Parallel Robot

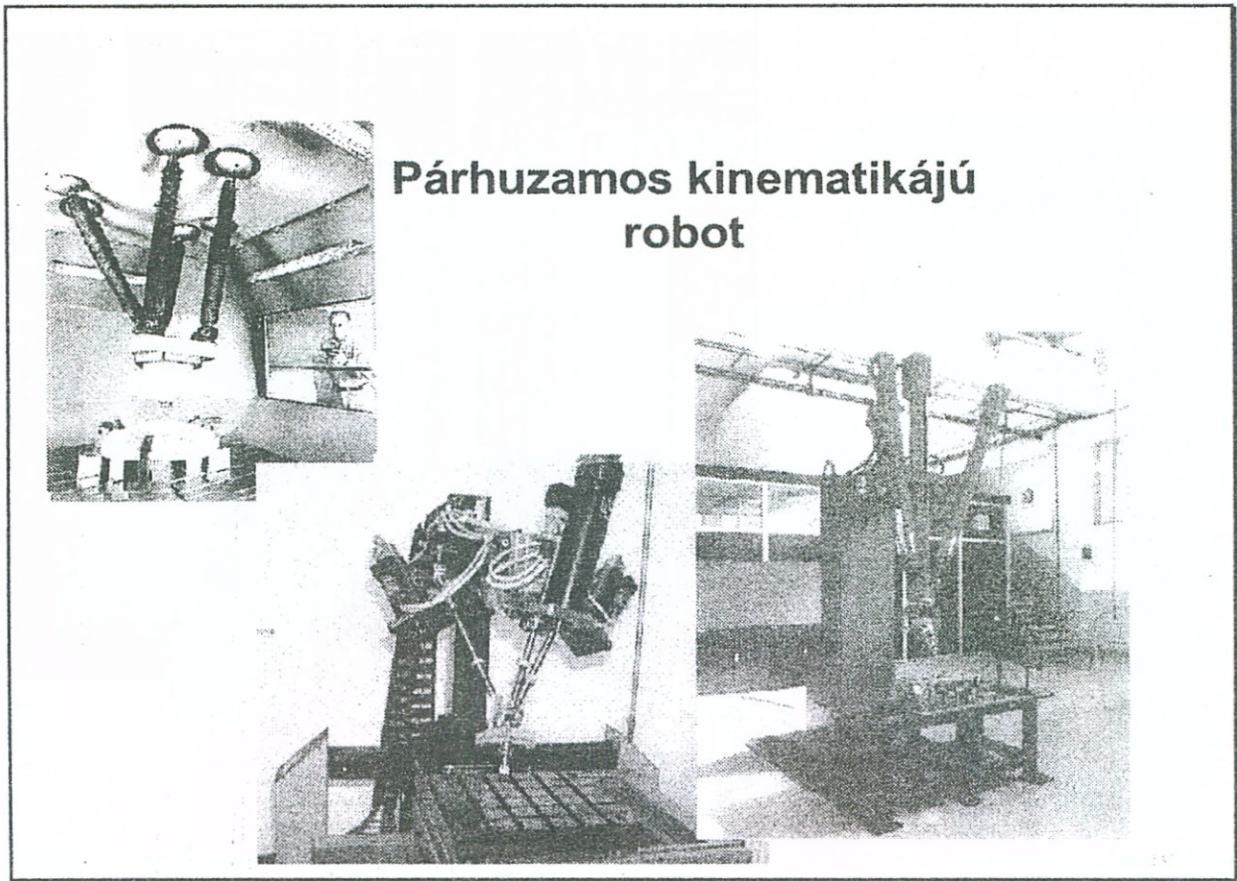
Koncepció



Struktúra

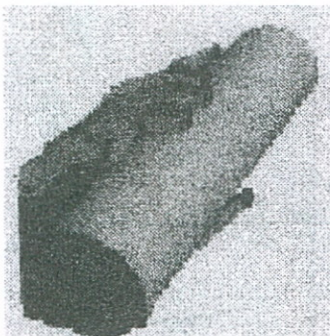


Munkatér



A mesterséges intelligencia alapjai

## Effektorok



A robotmechanizmus és a környezete közötti kapcsolatot biztosító aktív elem. Következő típusai lehetnek:

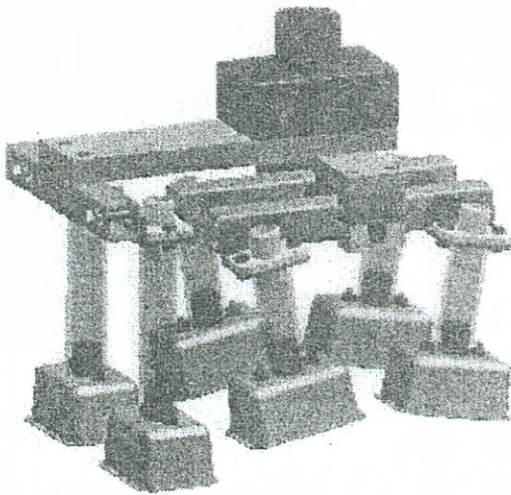
- A) Megfogók
- B) Szerszámok



## A megfogók energiaforrásai

- Elektronika
- Pneumatika
- Hydraulika
- Mágnesség
- Vákuum
- Egyéb

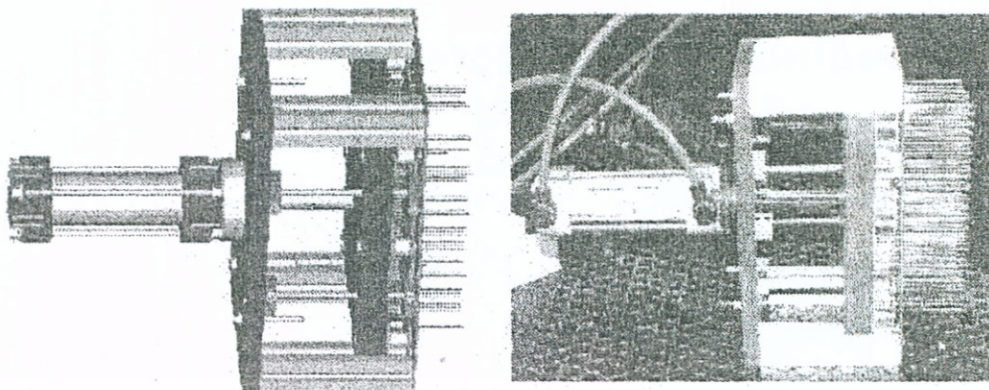
## Vákuum megfogó



Nem szilárd tárgyak  
megfogására szolgáló  
megfogószerkezet

A mesterséges intelligencia alapjai

## “Egyéb” erőforrással rendelkező megfogó



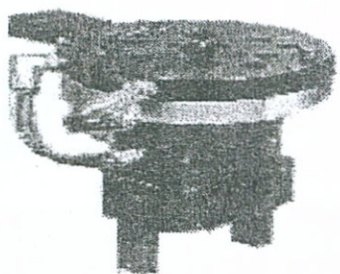
Nem szilárd tárgyak megfogására szolgáló megfogó

Dr. Szalay Tibor

24

A mesterséges intelligencia alapjai

## Hagyományos megfogószerkezet



Ez a leggyakrabban használt konstrukció nem más, mint két, azonos síkban mozgó, egyszabadságfokú ujj (pofa). Egyszerű feladatok megoldására azonban kiválóan alkalmas.

Dr. Szalay Tibor

25



## Emberi kezet mintázó megfogótípusok

Az emberi kéz a négy ujjban, a hüvelykujjban és a tenyérben mindösszesen 22 szabadságfokot tartalmaz. Ennek mesterséges megalkotása egy két, soros szabadságfokú rendszeren párhuzamosan elhelyezett öt 4 szabadságfokú, soros rendszert (robotot) eredményez. Több ilyen robot "kezet" fejlesztettek az elmúlt években, mind több "emberi" jelleget biztosítva a konstrukcióknak a szabadságfokok számának növelésével, illetve érzékelők alkalmazásával.

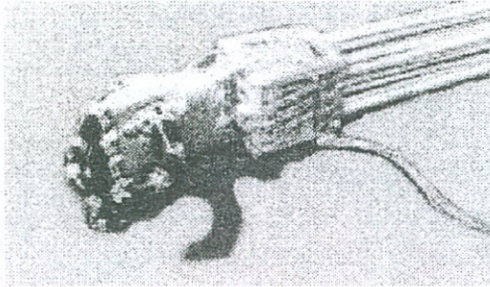
## Stanford/JPL Hand



Kenneth J. Salisbury a doktori dolgozata keretében fejlesztette ki az első, emberi ujjakat mintázó megfogóját, ami azóta a robotkezekkel foglalkozó kutatók kiindulása mindenütt. Három darab 3 izból álló (3 szabadságfokú) ujjból áll, melyeknek mozgásterjedelme nagyobb az emberi ujjakénál.

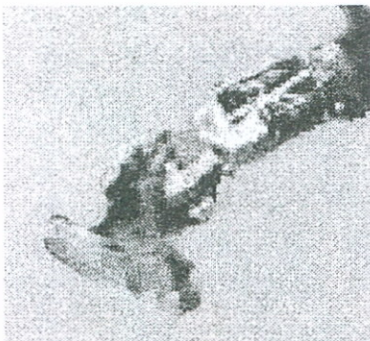


## Utah/MIT Hand



A nyolcvanas évek közepén keletkezett ez a konstrukció, amely már négy, egyenként négy szabadságfokú ujjat tartalmaz.

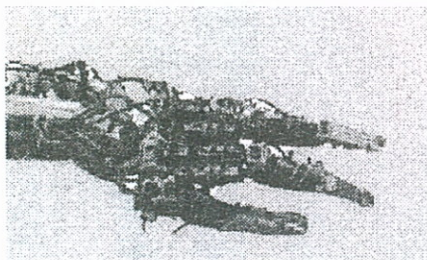
## Belgrade/USC Hand



Szintén a nyolcvanas évek terméke ez a kifejezetten antropomorf, négy ujjú plusz egy szembenálló hüvelykujjú kéz. Az ujjak mozgását összekapcsolták közös hajtással mozgatva, ezáltal csökkentették a motorok számát, és a vezérlési nehézségeket. Neurális hálózatos szoftver segítségével képes tanulni a megfogást.

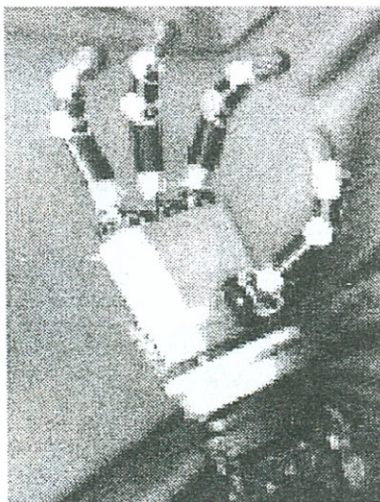


## Omni-Hand



A NASA-ban fejlesztették ki a jelenleg leginkább antropomorf kezét, amely a később bemutatandó Robonaut kézhez vezetett, és érzékelők hozzáadásával növelték a képességeit.

## Hirzinger Hand



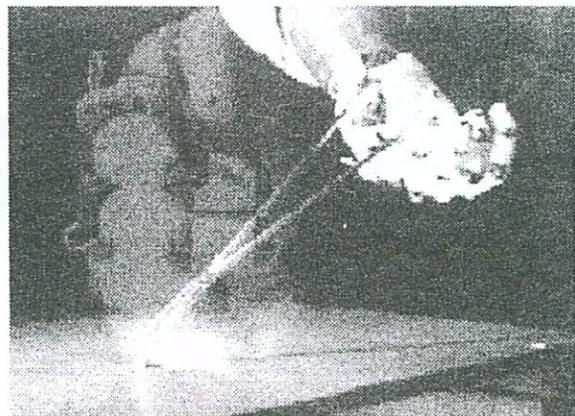
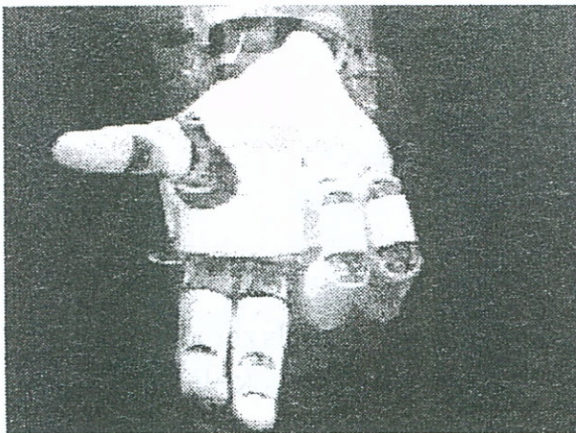
Ez a konstrukció kétszer akkora, mint egy emberi kéz, de más tekintetben szintén ahhoz hasonló mozgásokkal rendelkezik. A legösszetettebb szenzorrendszerrel szerelték föl, ujjanként 28 érzékelőt tartalmaz.

## Robonaut Hand Conceptual Model



Még nem megvalósított terv a NASA-nál az emberi kézhez (méretben is) leginkább hasonló konstrukció, amely 5 egyenként 12 szabadságfokú ujjal rendelkezik. Bár érzékelő rendszere nem olyan fejlett, mint a Hirzinger kézé, ujjain 5-5 szenzor biztosítja az érintési információkat.

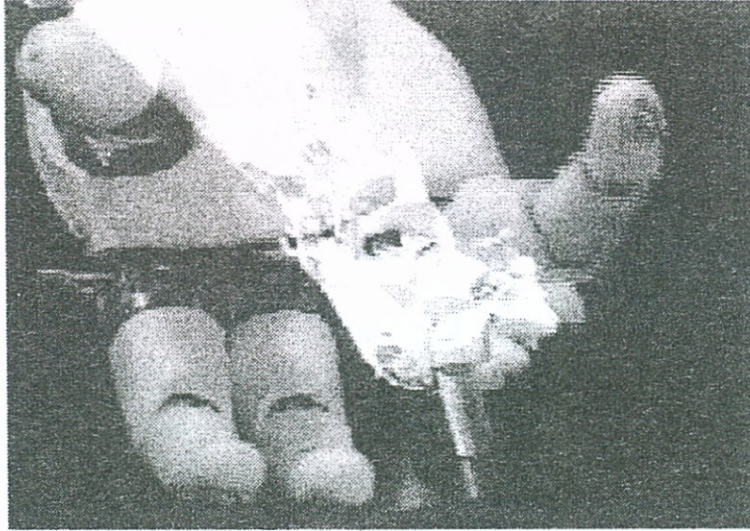
## Megvalósítás - Kéz





A mesterséges intelligencia alapjai

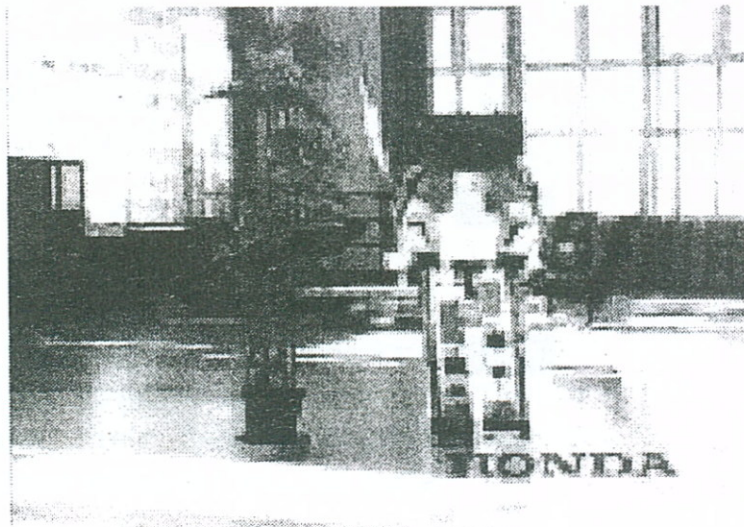
## Megvalósítás - Kar



Dr. Szalay Tibor

A mesterséges intelligencia alapjai

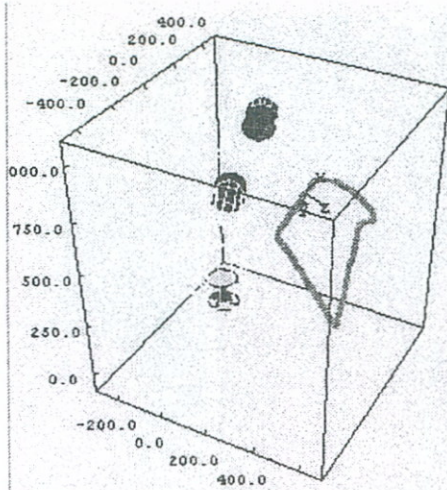
## Asimo



Dr. Szalay Tibor

35

## A koordináta rendszerek

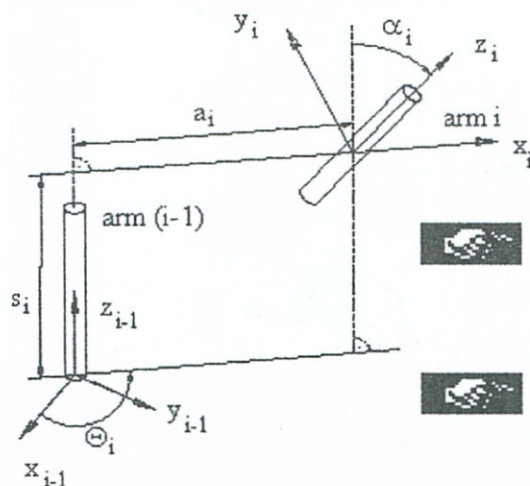


- világ koordináta rendszer
- csukló koordináta rendszer
- szerszám koordináta rendszer

### Probléma

A szerszám és a világ vagy a csukló koordináta rendszerek közötti geometriai transzformáció meghatározása

## Denavit-Hartenberg transzformáció



Az aktuális robotkar pozíciójának és orientációjának meghatározása az előző kar koordináta rendszerében:



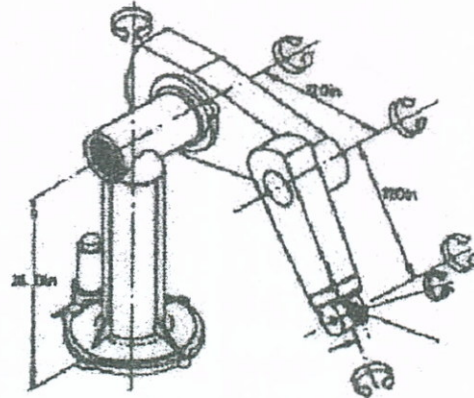
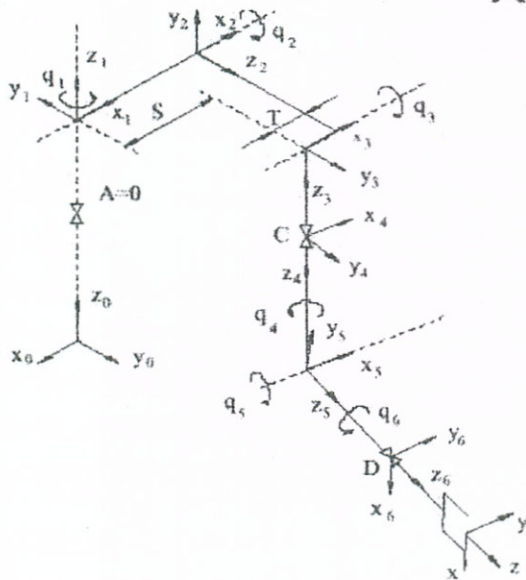
Az (i-1)-edik kar z tengelye legyen párhuzamos az i-edik kar mozgástengelyével



Az i-edik kar x tengelye legyen merőleges az i-edik és az (i-1)-edik karok z tengelye által meghatározott síkra







## A PUMA robot



## A transzformációk

Mindössze négy geometriai transzformáció elegendő egy robotkar pozíciójának és orientációjának meghatározására az előző kar koordinátarendszerében.

-   $\Theta_i$  elforgatás  $z_{i-1}$  körül miközben  $x_{i-1}$  párhuzamos marad  $x_i$ -vel,
-   $S_i$  lineáris eltolás parallelly with  $z_{i-1}$ -gyel párhuzamosan  $z_{i-1}$  és  $x_i$  metszéspontjáig,
-   $a_i$  lineáris eltolás  $x_i$  vel párhuzamosan amíg a két koordinátarendszer középpontja azonos nem lesz,
-   $\alpha_i$  elforgatás  $x_i$  körül amíg a  $z$  tengelyek párhuzamosak nem lesznek

### A transzformációs mátrix

$$D_{i-1,i} = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \cdot \sin\theta_i & -\sin\alpha_i \cos\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \sin\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & s_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Az eredő mátrix n karú robot esetén

$$E = D_{0,1} * D_{1,2} * D_{2,3} * \dots * D_{n-1,n}$$

### Egy pont ranszformációja

A világ koordinátarendszer egy pontjának a szerszám koordinátarendszerben a következő transzformáció segítségével adhatjuk meg a koordinátáit:

$$\begin{bmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{bmatrix} = D_{i-1,i} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$



## Az inverz transzformáció

A robot programozás alapproblémája a következő:

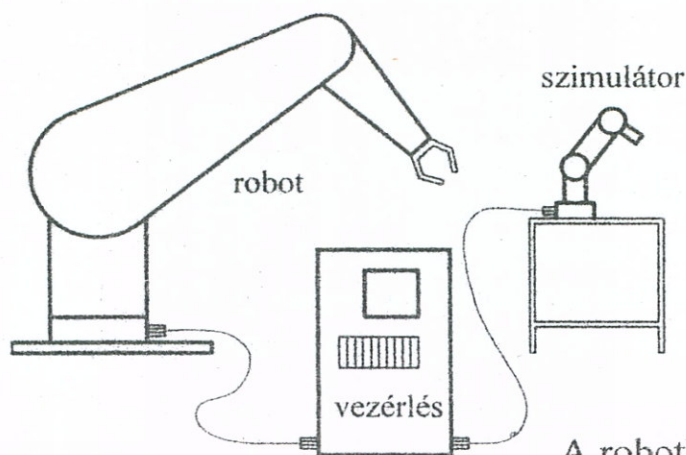
A tárgy egy pontja a szerszám koordinátarendszerben adott, és azt kell a világ koordinátarendszerbe transzformálni, vagyis az előzőekben felírt számítás inverze a feladat. Ennek az inverz transzformációnak azonban több megoldása is lehetséges (tehát szinguláris pontjai vannak a robot munkaterének).

Néhány konfigurációs definícióval egyértelművé tehető a megoldás (jobb illetve bal kezes konfiguráció, alsó vagy felső állás).

## A robot programozás három formája

- Betanítás, és a feladat ismétlése
- Kódrendszerben történő programozás
- Magasszintű programnyelven történő programozás (az utasítások az emberi nyelv azonos értelmű szavainak felelnek meg)

## A robot betanítása



A robotkar mozgatásával  
 Szimulátor mozgatásával  
 "Teach box" alkalmazásával

## Robot programozás

**Kódrendszer alkalmazásával (NC programozáshoz hasonlóan).**

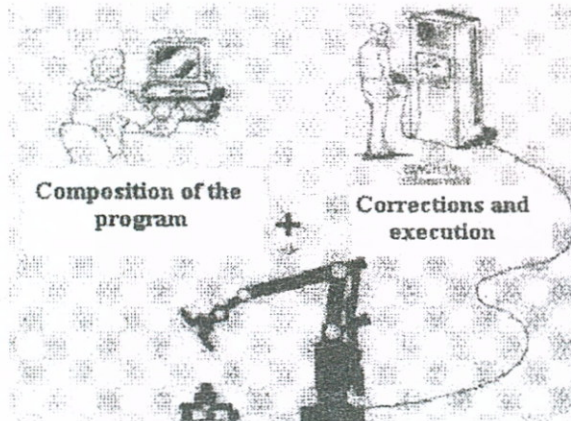
NC mondat sorszáma	Program kód	Paraméterek
N100	G0	x,y,z, $\alpha$ , $\beta$ , $\gamma$

**Robotnyelv alkalmazásával.**

WAVE, AL, VAL, PAL, AML, RAIL, MCL, RCL



## Számítógéppel Segített Robot Programozás Computer Aided Robot Programming (CARP)



A robotpályák meghatározása  
(CAD/CAM software segítségével)

Posztprocesszálás (a robot program generálása)

A program áttöltése

Futtatás

## A VAL programnyelv szerkezete

### Interpreter utasítások (operációs rendszer)

calibráció, szerkesztés, tanítás, mozgás, mentés vagy betöltés

### Program utasítások

konfiguráció, mozgás, kommunikációk, aritmetikai műveletek, ciklusok és szubrutinok, memória regiszterek, paraméterek, változók, pontok

## Program utasítások a VAL-ban

### Mozgási utasítások

<b>MOVE P1</b>	<b>APPRO P1,50</b>
<b>MOVES P1</b>	<b>DEPART 50</b>
<b>MOVE P1 VIA A1</b>	<b>SPEED 200</b>
<b>DMOVE P1 &lt;x,y,z&gt;</b>	

## Program utasítások a VAL-ban

### Konfigurációs utasítások

**LEFTY**

**RIGHTY**

**DEFINE FRAME1=FRAME(A1,A2,A3)**

**DEFINE PATH1=PATH(P1,P2,P3,P4...)**

**SHIFT <x,y,x>**



## Program utasítások a VAL-ban

### Kommunikációs utasítások

**OPEN**

**CLOSE**

**SIGNAL 1,2,-5**

**WAIT 2**

**IFSIG -2 THEN GOTO 100**

## Program utasítások a VAL-ban

### Program szervezés

**GOTO 100**

**IF A>5 THEN GOTO 100**

**DO <commands> UNTIL A>5**

**DELAY 10**

**CALL Name**

**STOP**

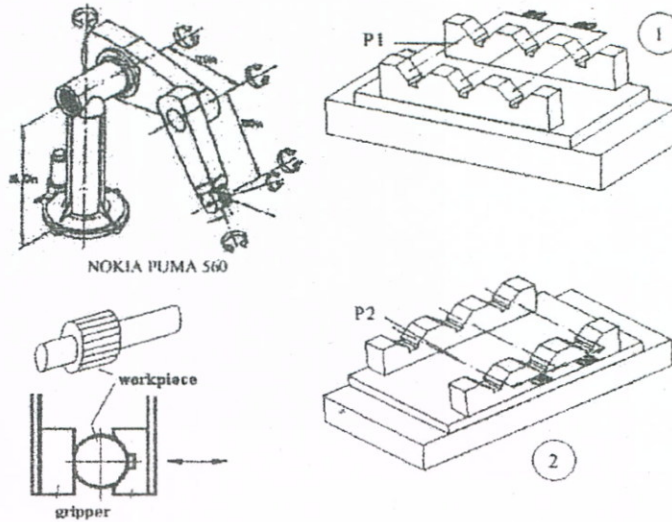
**PAUSE**

**RESUME**

**CONTINUE**

**RETURN**

### Példa (Mátyási Gyula docens)



Dr. Szalay Tibor

52

### A program (Mátyási Gyula docens)

20 WAIT 3	MOVES P1
SPEED 500	CLOSE
LOCATE P11=P1	DELAY 0.5
LOCATE P22=P2	APPRO P1,100
SET I=1	WAIT 2
10 APPRO P1,100	SIGNAL -2
OPEN	SPEED %100
WAIT 1	APPRO P2,100
SIGNAL -1	SPEED % 10
SPEED % 10	MOVES P2

Dr. Szalay Tibor

53



## A program (Mátyási Gyula docens)

```
OPEN                                GOTO 10
APPRO P2,100                        100 LOCATE P1=P11
SPEED % 100                         LOCATE P2=P22
SET I=I+1                            SIGNAL 1
IF I=4 THEN GOTO 100                SIGNAL 2
SHIFT P1=0,80,0                     GOTO 20
SHIFT P2=0,80,0                     STOP
```

## 7.5. Robotok alkalmazása

### A robotok alkalmazásának ösztönzői

1. Technológiai (pontosság, megbízhatóság, minőség)
2. Gazdasági (termelékenység, minőség, magas személyi költségek)
3. Biztonság (veszélyes környezet, anyagok, műveletek)

A mesterséges intelligencia alapjai

## Robotok alkalmazási tulajdonságai az ISO 9283 alapján

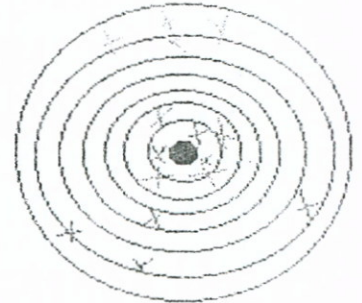
1. Szabadságfokok száma (csuklók száma és típusa)
2. Munkatér alakja és nagysága
3. Mechanikai merevség és engedékenység
4. Abszolút és ismétlési pontosság
5. Maximális terhelés
6. Sebesség
7. Programozási tulajdonságok (interpolációk, programszervezés)

A+,R+

A+,R-

A-,R+

A-,R-

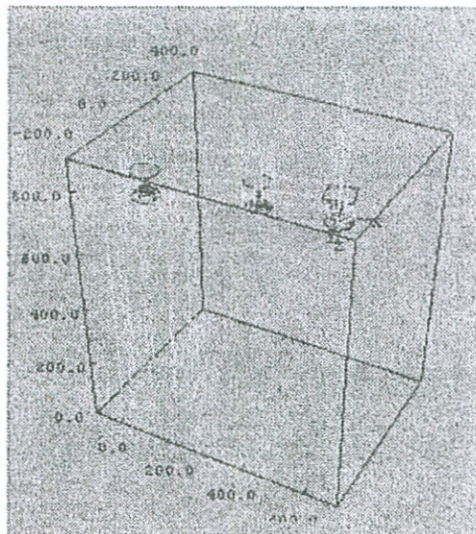


Dr. Szalay Tibor

56

A mesterséges intelligencia alapjai

## A SCARA robot tulajdonságai



Függőleges irányban nagy merevség

Vízszintes irányban engedékeny

Egyszerű vezérelhetőség

Gyors mozgások

Olcsó

Dr. Szalay Tibor

57



A mesterséges intelligencia alapjai

## Gyártási alkalmazás (hegesztés)



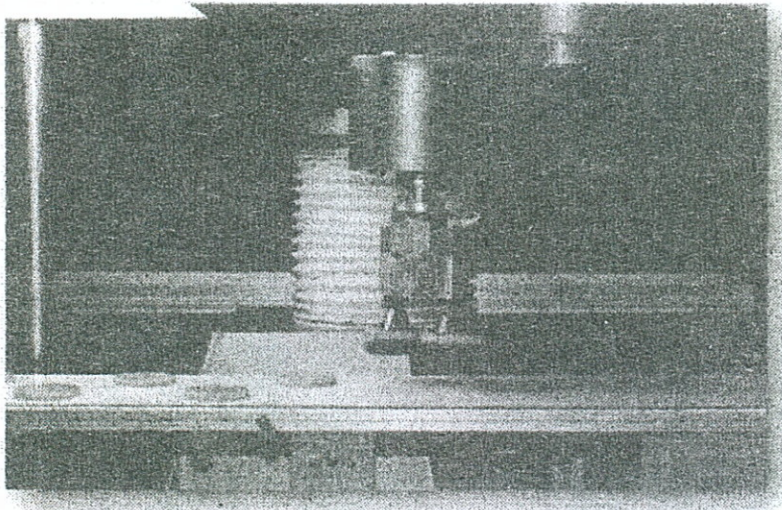
Dr. Szalay Tibor

58



A mesterséges intelligencia alapjai

## Manipulációs alkalmazás (McDonald's)



Dr. Szalay Tibor

59



A mesterséges intelligencia alapjai

## Alkalmazás a Biotechnológiában

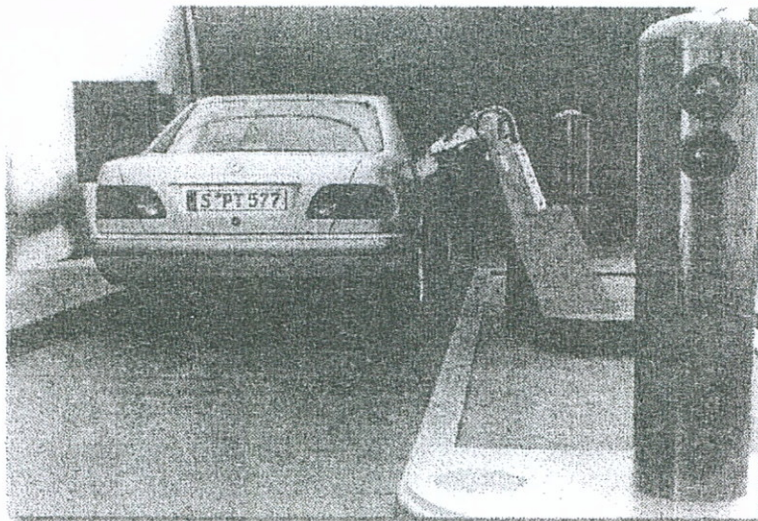


Dr. Szeley Tivor

60

A mesterséges intelligencia alapjai

## Autómatikus tankolás

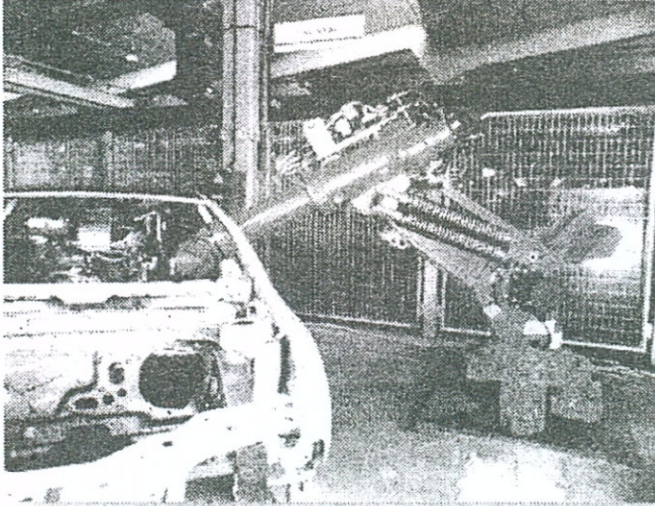


Dr. Szeley Tivor

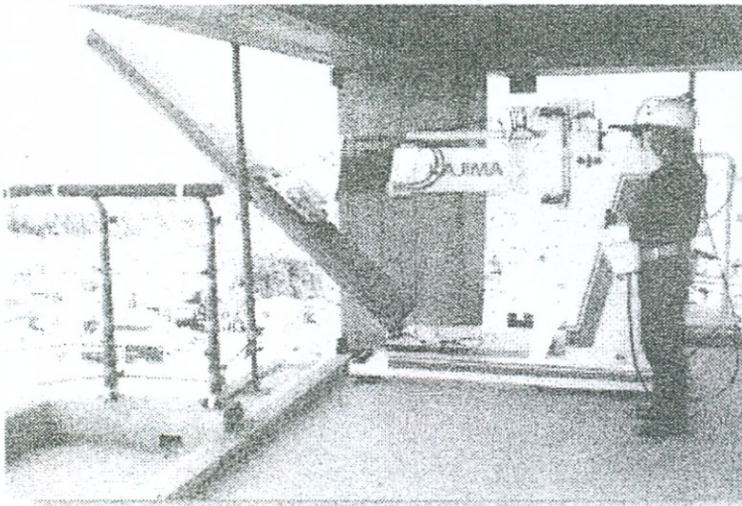
61



### Gyártási alkalmazás (szerelés)

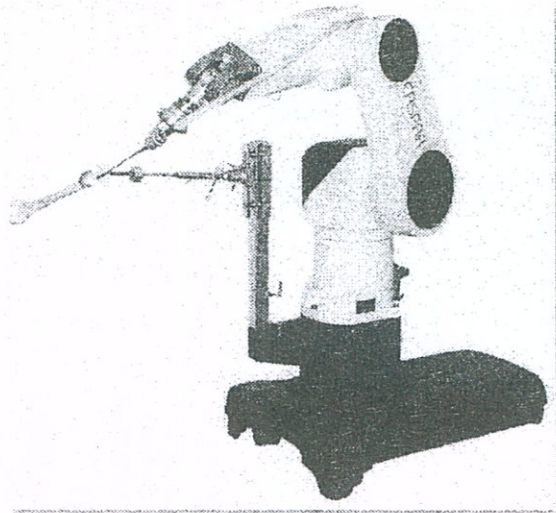


### Nehéz darabok manipulálása



A mesterséges intelligencia alapjai

## Orvosi alkalmazás

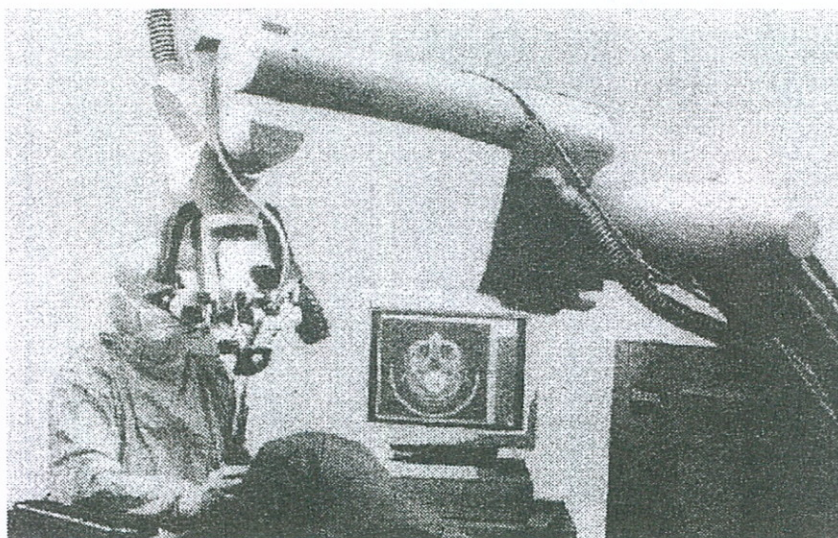


Dr. Szalay Tibor

64

A mesterséges intelligencia alapjai

## Sebész robot



Dr. Szalay Tibor

65



A mesterséges intelligencia alapjai

## Alkalmazás a gyógytornában

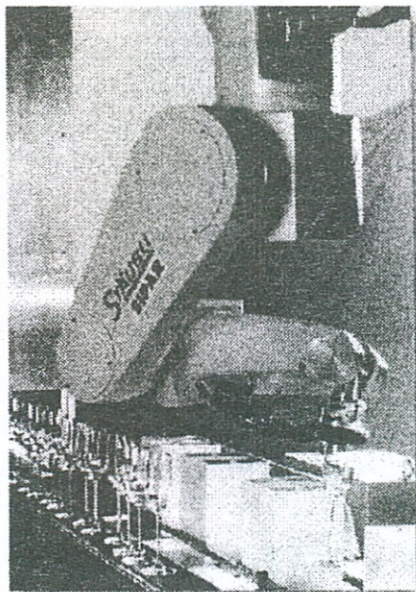


Dr. Szalay Tibor

66

A mesterséges intelligencia alapjai

## Csomagolóstechnológiai alkalmazás



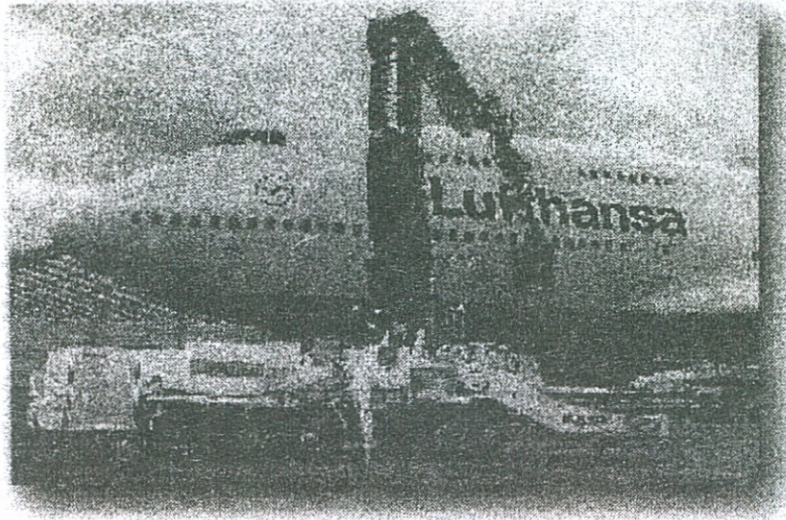
Dr. Szalay Tibor

67



A mesterséges intelligencia alapjai

## Repülőgép tisztítás, festés

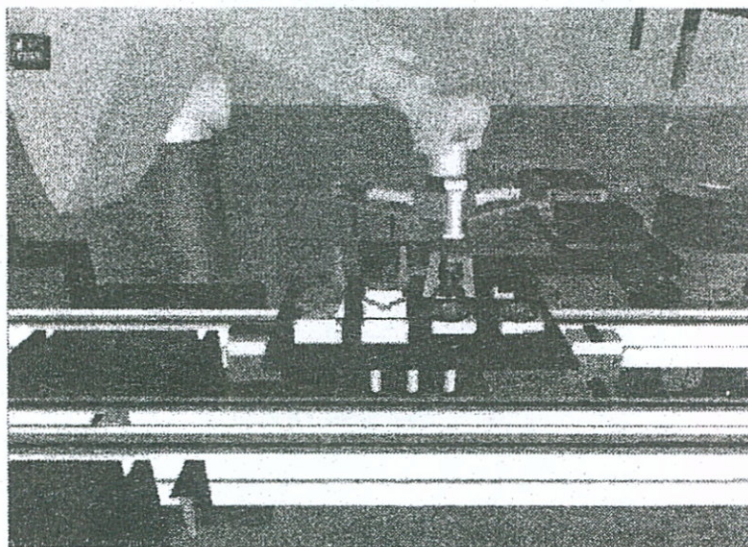


Dr. Szalay Tibor

68

A mesterséges intelligencia alapjai

## Robotos szerelés

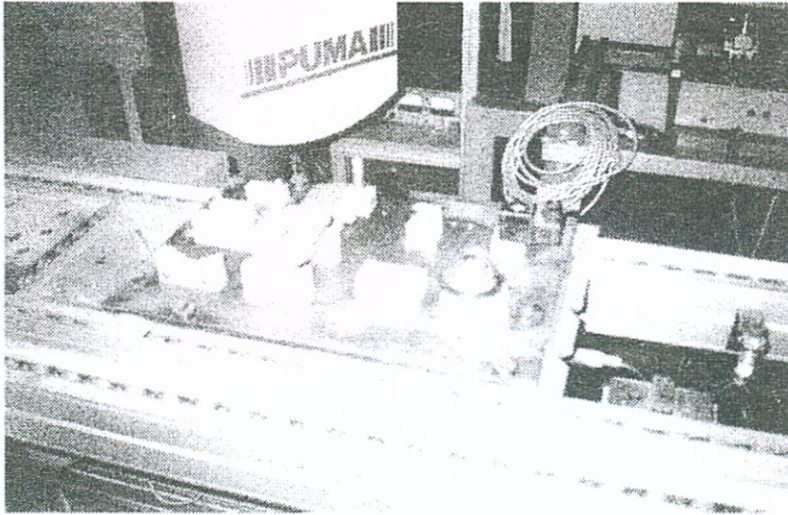


Dr. Szalay Tibor

69

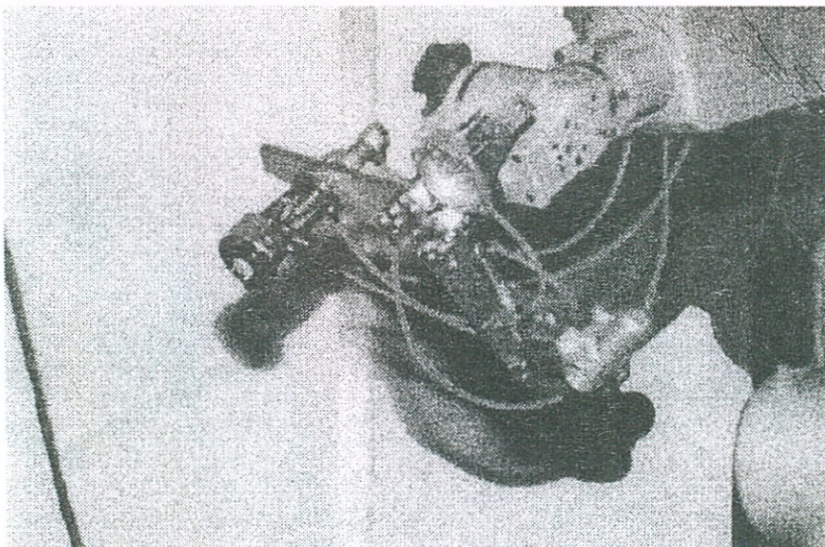


## A paletta



A mesterséges intelligencia alapjai

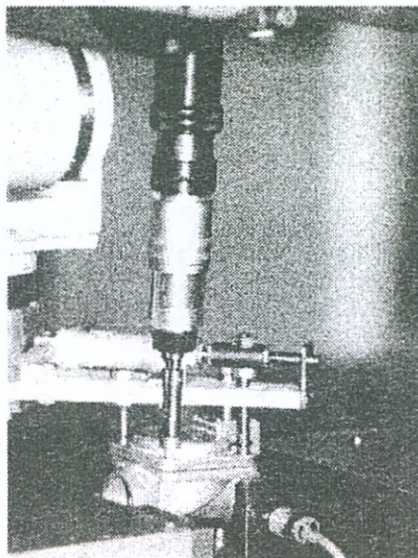
## A megfogó



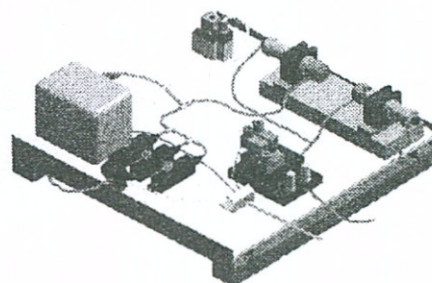
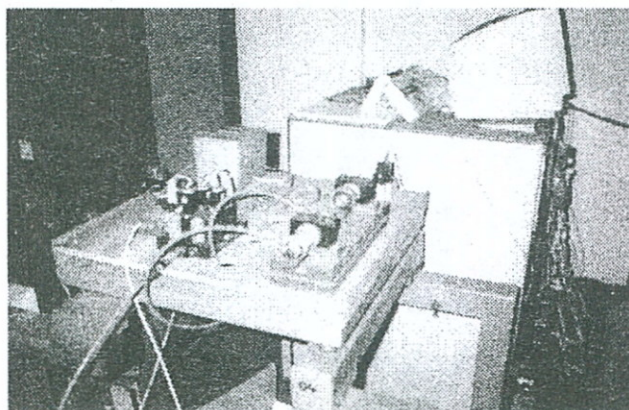
Dr. Szalay Tibor

71

### A csavarozó



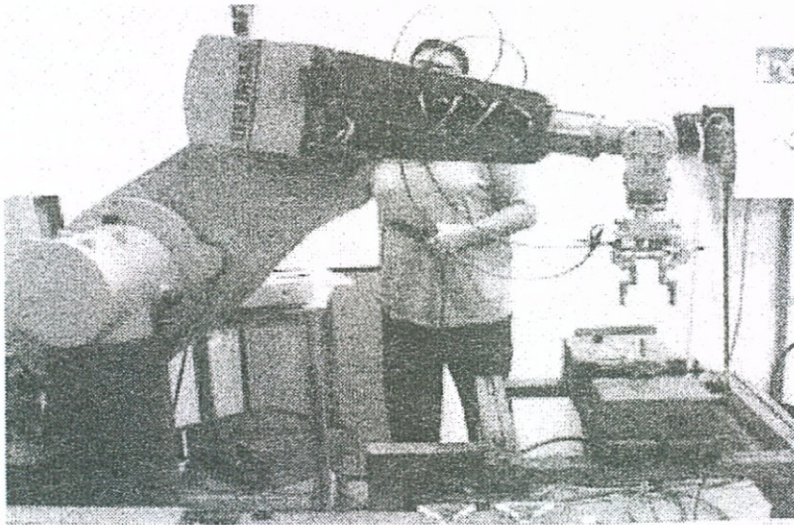
### Nyomás ellenőrzés





A mesterséges intelligencia alapjai

## Nyomás ellenőrzés



Dr. Szalay Tibor

74





## 8. INTELLIGENS GYÁRTÁS

Napjaink gyártási környezete egyre összetettebb, a gyártórendszereket érő hatások miatt a bizonytalansági faktor megnőtt. Ennek okai a következők:

- új feladatok gyors egymásutánisága,
- a közvetlen környezetben bekövetkező események hatása,
- a gyártási rendszerekben működő és a hozzájuk kapcsolódó kommunikációs csatornák számának növekedése,
- fogyasztói igények sokrétűsége,
- a nagyszorozatok csökkenő dominanciája, az egyedi gyártás növekvő szerepe,
- az információs technológia gyors fejlődése,
- a fogyasztók bekapcsolása a termelési folyamatba (pl. tervezés).

A fent említett jelenségek már napjainkban is érzékelhető hatással vannak a gyártórendszerek struktúrájára, illetve működésére. A következő évtizedekben forradalmi változások várhatóak a gyártórendszerek fejlődésében, melynek során előtérbe kerülnek a kisebb méretű, de jóval nagyobb rugalmasságú, autonóm, egymással együttműködő gyártóegységek. A jelenlegi, erősen hierarchikus felépítésű rendszerek helyébe innovatívabb, rugalmasabb működést biztosító elosztott struktúrájú szervezetek kerülnek.

Mindeddig a *Számítógéppel Integrált Gyártás* (CIM) koncepciója révén várták a megoldást a dinamikusan változó környezet kihívásaira. A hagyományos CIM rendszerek felépítése általában erősen hierarchikus szerkezetű. Napjainkra kiderült, hogy az ilyen struktúra nem minden esetben alkalmas a felsorolt problémák kezelésére. A CIM-et telepítő cégek nagy befektetés mellett, hosszú kifutási idejű és az erős centralizáció miatt általában rendkívül merev gyártási struktúra mellett kötelezik el magukat. A magas szintű automatizáltság mellett a rendszereket jellemző merevség lehetetlenné teszi alkalmazkodásukat a folyamatosan változó környezethez.

A probléma megoldásának érdekében intenzív kutatások folynak új gyártási struktúrák kialakítására. Ilyen új kutatási irányzatok olyan gyártórendszerek kifejlesztését célozzák, amelyek autonóm, intelligens, rugalmas, elosztott, egymással kooperáló, a környezethez dinamikusan és rugalmasan alkalmazkodó egységekből, állnak. Japánban dolgozták ki a holonikus gyártórendszer modellt, míg európai (elsősorban német) példa a fraktál vállalat modellje.



## ELŐADÁS

Gépgyártástechnológia II.

BME GTT

**Intelligens gyártás**

- Holonikus gyártás
- Fraktál vállalat
- MI módszerek:
  - Neurális hálózatok
  - Fuzzy logika
  - Genetikus algoritmusok
  - Szakértő rendszerek

Dr. Szalay Tibor

Gépgyártástechnológia II.

BME GTT

**Az Intelligens Gyártórendszer (IGYR)  
Intelligent Manufacturing System (IMS)  
paradigma**

- Egyedi gyártás növekvő igénye
- Integráció fokozódása
- Átfutási idők csökkentése
- Minőségi jellemzők fontossága
- Gazdasági és műszaki jellemzők összefonódása

Dr. Szalay Tibor



## Az Intelligens Gyártórendszer (IGYR) Intelligent Manufacturing System (IMS) paradigma

- Tervezés, gyártás és piac globalizálódása
- A fenntartható fejlődés (az erőforrások takarékos és „zöld” kihasználása)
- Ember nélküli gyárak
- Mesterséges intelligencia technikák alkalmazása

Dr. Szalay Tibor

## A problémák megoldása

Soha nem tapasztalt összetettség

Bizonytalan faktorok számának növekedése



Dirakt vezérlés számára kezelhetetlen

➔ Holonikus gyártás

➔ Fraktál vállalat

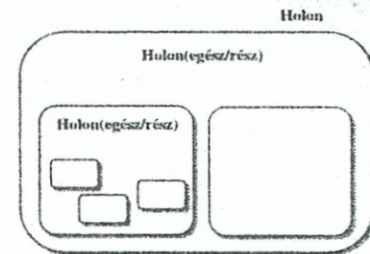
Dr. Szalay Tibor



## 8.1. Holonikus gyártási modell

### Holonikus gyártás

Holosz (görög) = egész  
 -on (görög) = rész



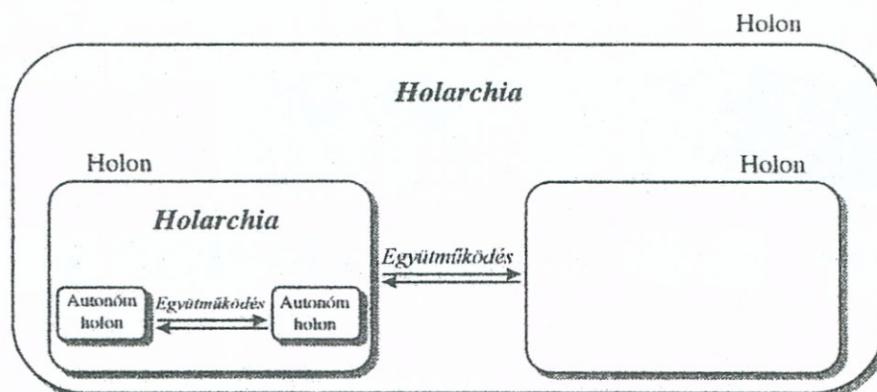
**Holon**  
 (Arthur Koestler 1967)

Eredet:

- Herbert Simon → órasmesterek  
 Tempus – alapelemekből      Horus – részegységekből
- Élő és társadalmi szervezetek hierarchiája  
 Minden ami egész egy nagyobb egység része és fordítva

Dr. Szalay Tibor

### Holarchia



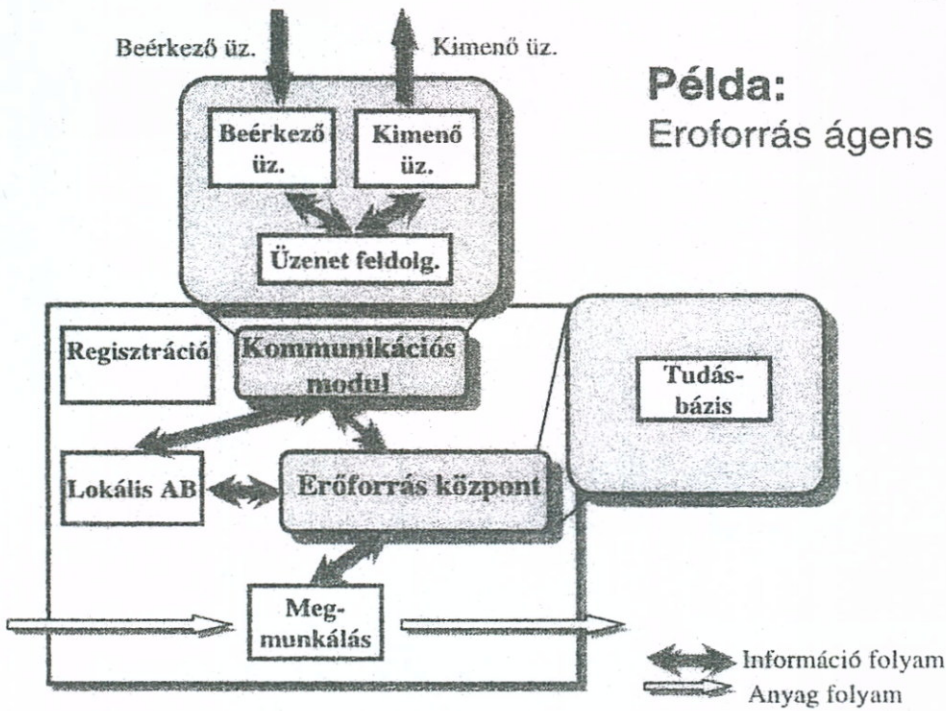
A holonikus gyártórendszer autonóm, intelligens, rugalmas, elosztott, egymással kooperáló egységekből (holonokból, ágensekből) áll.

Dr. Szalay Tibor



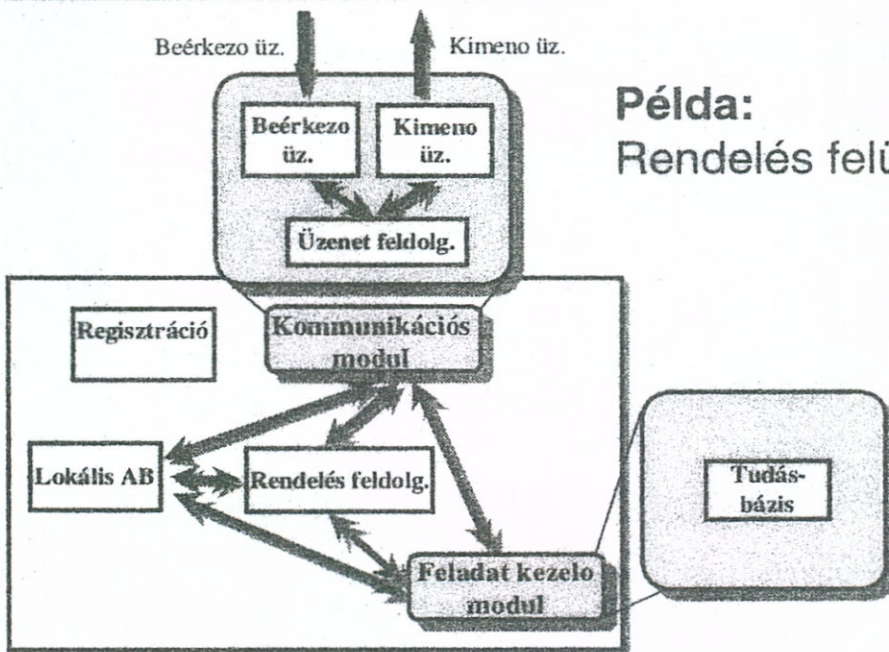


**Példa:**  
Eroforrás ágens



Dr. Szalay Tibor

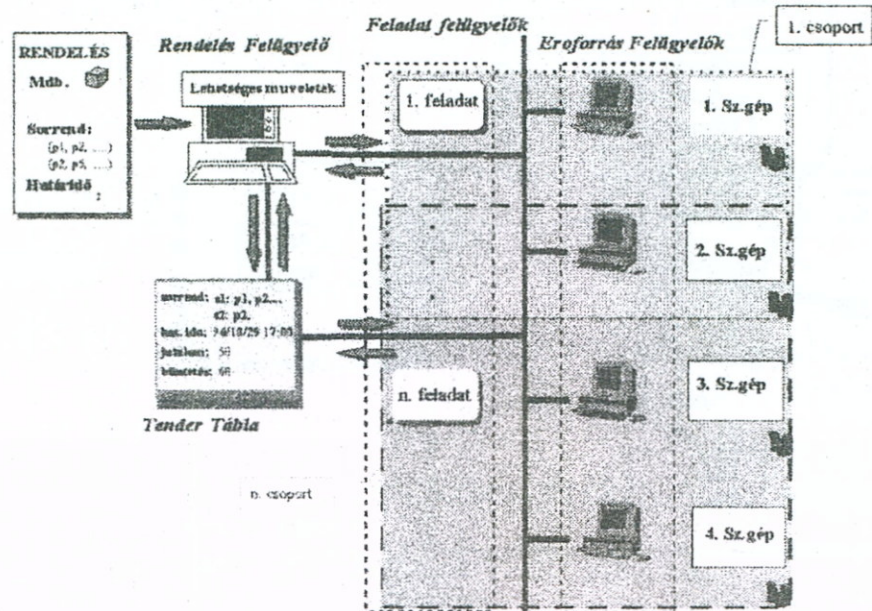
**Példa:**  
Rendelés felügelyelo ágens



Dr. Szalay Tibor

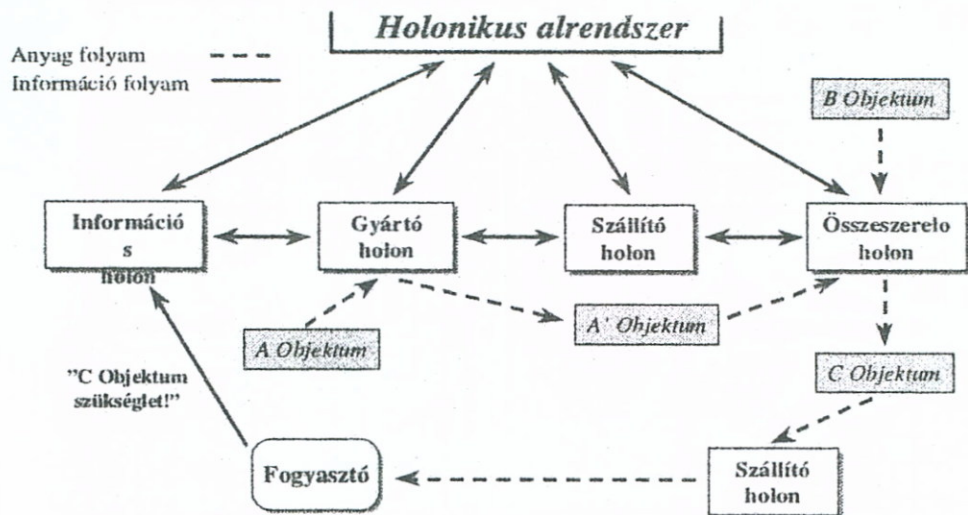


**Példa: Véletlenszerűen érkező megrendelések kezelő elosztott gyártórendszer**



Dr. Szalay Tibor

**Holonikus Gyártórendszer Struktúra**



Dr. Szalay Tibor



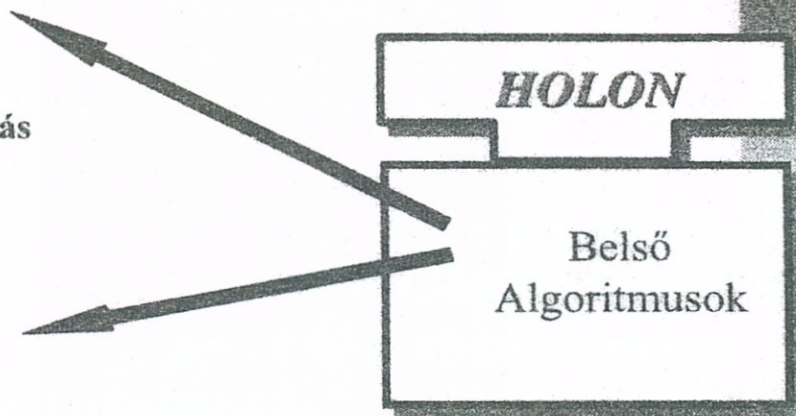
### Holonikus tulajdonságok I.

**Autonómia:**

- Önütemezés
- Önszabályozás
- Önjavítás
- Önbeállítás

**Kooperáció:**

- Tárgyalás
- Kommunikáció

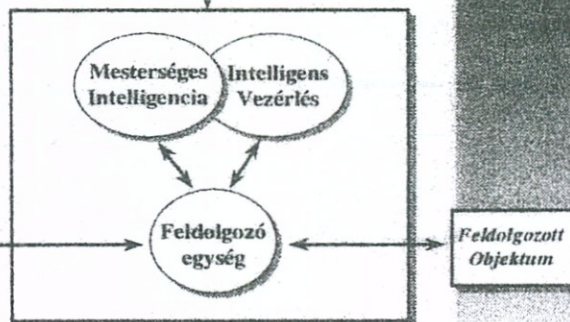


Dr. Szalay Tibor

### Holonikus tulajdonságok II.

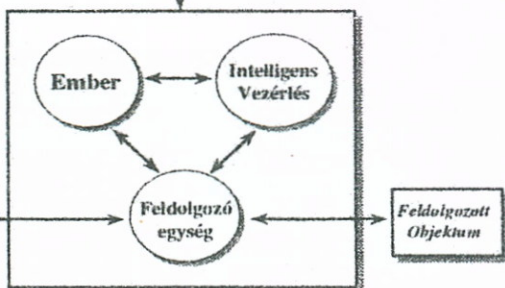
Az ember könnyen beilleszthető, hiszen autonóm és együttműködésre képes

Holarchia



Embertől független

Holarchia



Embertől függő

Dr. Szalay Tibor

## 8.2. Fraktál vállalat modell

### Fraktál vállalat

A holonikus gyártórendszer koncepcióhoz hasonló (az előzőt a japánok vezették be, míg a fraktál gyártás európai elképzelés.

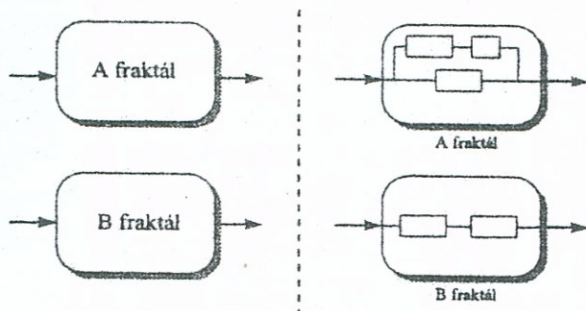
(Prof. Warnecke)

Itt is intelligens, elosztott, autonóm, rugalmas és egymással együttműködő egységek rendszeréről van szó.

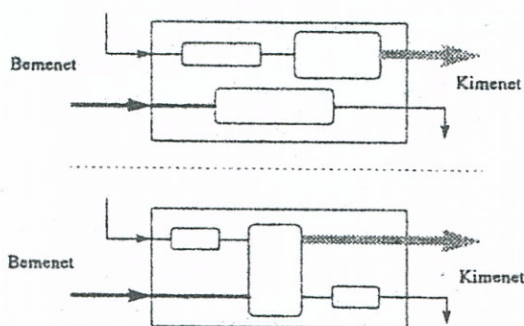
Fraktál tulajdonságok: Hasonlóság  
Önszerveződés  
Dinamizmus és vitalitás

Dr. Szalay Tibor

### Hasonlóság



Egymásba ágyazott felépítés (fraktálok fraktálokban).



Hasonló feladatot ellátó fraktálok belső felépítése különbözhet.

Dr. Szalay Tibor



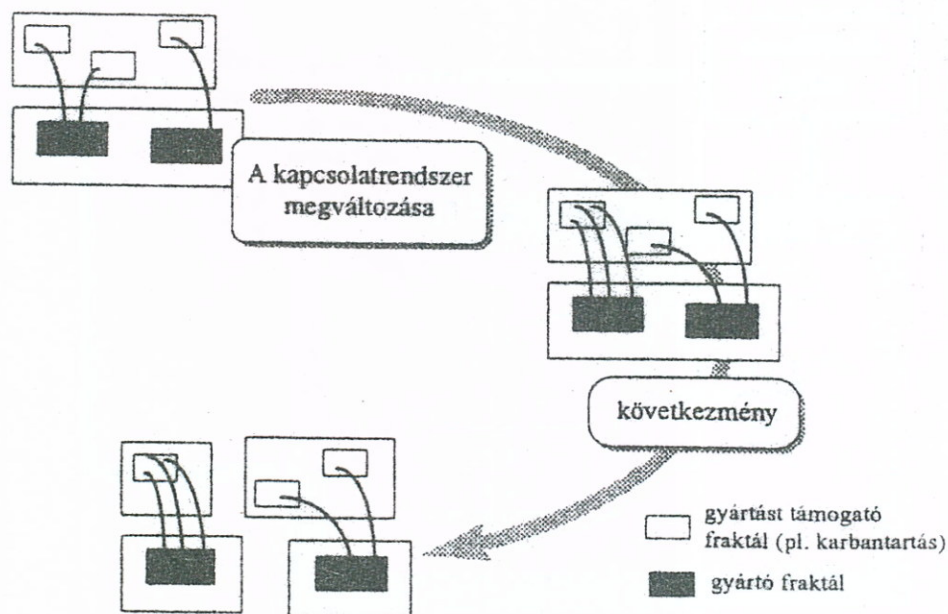
## Önszerveződés

- Fraktál önmagán belül dönt belső struktúrájáról és működéséről
- Nagymértékű decentralizáció
- Fraktálok közötti együttműködések kapcsolatok szerveződése (dinamikus struktúrálódás)

3M „post it” üzenőcédula ilyen körülmények között került kifejlesztésre

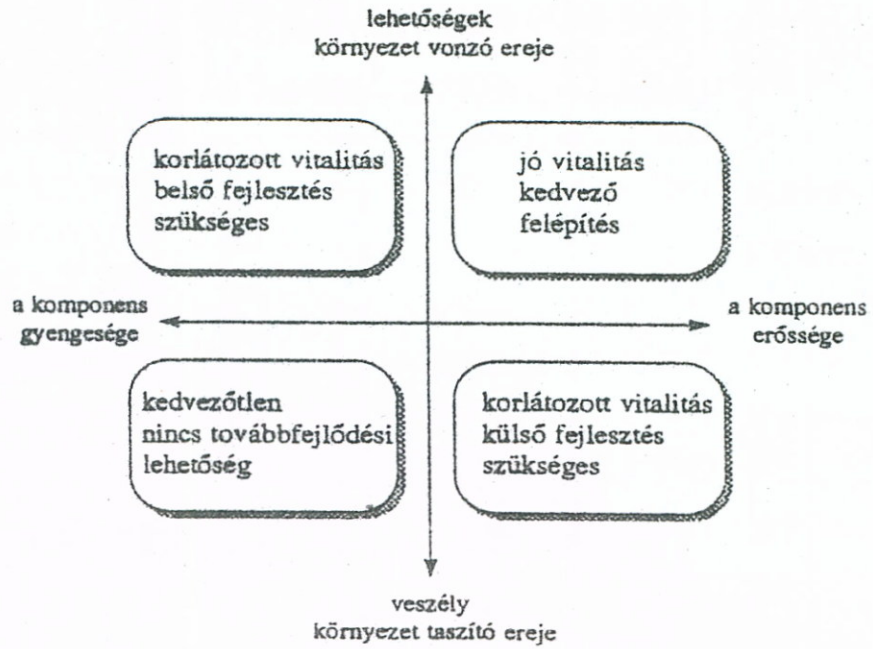
Dr. Szalay Tibor

## Dinamizmus és vitalitás



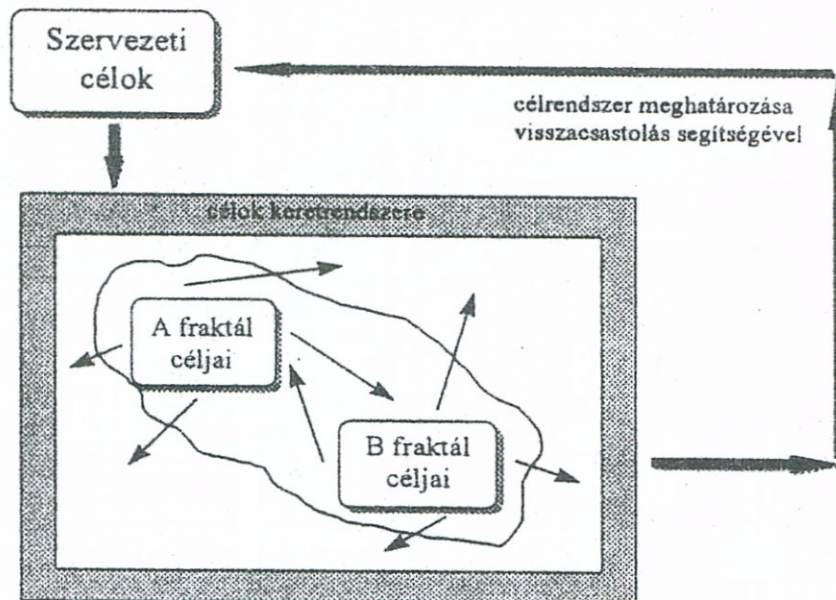
Dr. Szalay Tibor

### Vitalitás portfólió



Dr. Szalay Tibor

### Fraktál vállalat célrendszerének kialakítása



Dr. Szalay Tibor



ISBN 963 9625 35 3



9 789639 625358