

# Fogalom értelmezések II.

## Mikor célszerű neurális hálózatot alkalmazni?

- Ha a megoldandó problémával kapcsolatban gazdag adathalmaz áll rendelkezésre
- Ha a megoldáshoz szükséges szabályok ismeretlenek
- Ha a rendelkezésre álló adathalmaz nem teljes, hibás adatokat is tartalmazhat
- Ha sok összefüggő bemenő adat-, összefüggő kimeneti paraméter áll rendelkezésre

## A neurális hálózatok tervezése során megválasztandó paraméterek

### *Processzor szinten:*

- a processzor átviteli függvénye
- a bemenő jelek típusa

### *Hálózat szinten:*

- a hálózat topológiája
- rétegelt struktúrájánál a rétegek száma
- processzorok száma a különböző rétegekben
- processzorok típusa a különböző rétegekben

### *Tanítási szinten*

- tanító algoritmus
- tanítási paraméterek (pl.:  $\alpha$ ,  $\beta$ , ...)
- a tanítás megállásának feltételei

## A neurális hálózatok és a szakértői rendszerek jellemzőinek összehasonlítása

Neurális hálózatok	-	szakértői rendszerek
Nincs tudás-hozzáférési probléma		nehéz a tudás megszerzése
Dinamikus tudásábrázolás		statikus(abb) tudásábrázolás
Minta kiegészítő képesség		feltételezzük az adatok hibátlanságát és ellentmondás mentességét
Robosztus (nem érzékeny az adatvesztésre)		Érzékeny az adatvesztésre
Interpolálni képes		Nincs intuitív képessége
Többet „tudhat” mint ami az adatokból látszik		Legfeljebb olyan jó lehet, mint amire felkészítették
Nincs magyarázat adás		Részletes magyarázat kérhető
Nincs igazoló képessége		A döntéseket viszonylag könnyű igazolni.

## A lágy számítási modell értelmezése, tipikus formái

A lágy számítási modellek a biológiai információ-feldolgozás példáiból táplálkozó számítógépes információ-feldolgozó megoldások (paradigmák)

Főbb formái, területei:

1. Neurális hálózatok, ahol az idegrendszer működése a referencia
2. Genetikus algoritmusok, ami a természetben működő populáció- és relatív jószág alapú kereső módszer számítógépes megoldásával foglalkozik
3. Fuzzy logika, melyben az emberi (homályos) fogalmakkal megfogalmazott szabályokra és az un. tagsági függvényekkel jellemezhető igazságtartalmú változókra épül a számítógépes információ-feldolgozás

## A Fuzzy logika lényege és alkalmazásának alapfeltevése

- A rendszerek működését és vezérlését meghatározó törvények nyelvi eszközökkel

(szavakkal) írja le.

- Alapja a Fuzzy halmazelmélet.
- Bevezeti a részleges igazságtartalmat. Tagsági függvénnyel értelmezi az átmenetet a 100 százalékban igaz és 100 százalékban hamis között.
- A matematika, a számítástudomány és a villamosmérnöki tudományok határterületén helyezkedik el.
- A fuzzy értelmezéssel az emberi tudás megjeleníthető a technikában

### A Fuzzy logika alkalmazásának ígéretei

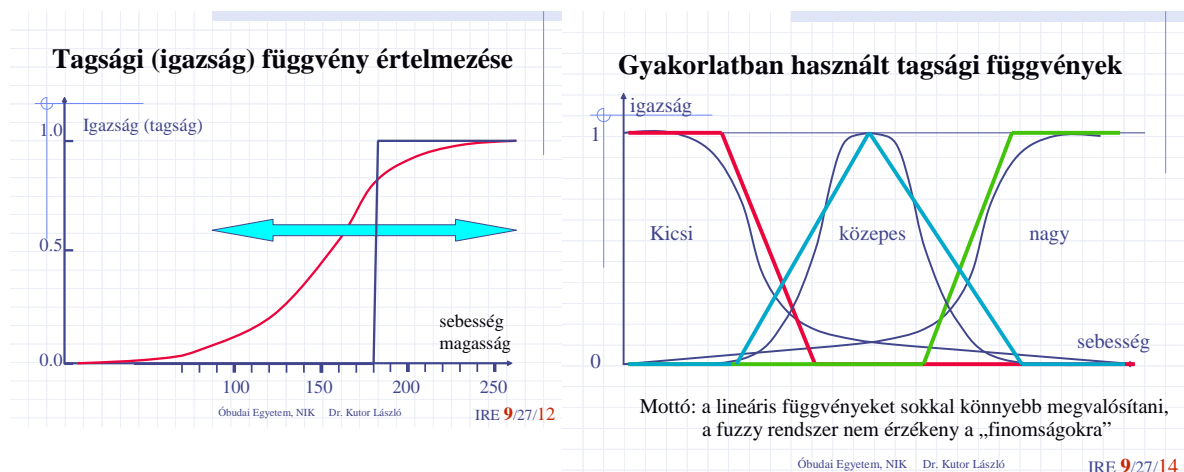
- Következetes és szilárd alapot ad a pontatlan és bizonytalan információ feldolgozáshoz.
- Interfészt biztosít az emberek által kedvelt nyelvi változók és a számítógépek mennyiségi változói között.
- Hidat képezhet az MI szimbólum feldolgozó megközelítése és a neurális hálózatok között.
- A hagyományos modellekkel szemben jelentősen egyszerűbb rendszer leírást tesz lehetővé.

### Mikor célszerű Fuzzy logikát használni?

- Összetett rendszerekben, ahol nehéz, vagy lehetetlen megfelelő rendszermodellt kialakítani
- Olyan rendszerekben, melyeket szokásosan emberi szakértő irányít, (emberek adják a bemeneteket vagy a szabályokat)
- Olyan rendszerekben melyek folyamatos, vagy közel folyamatos bemenetekkel és nem lineáris kimeneti válaszfüggvényekkel jellemezhetők
- Olyan rendszerekben, melyekben a pontatlanság és a homályosság a rendszer gyakori velejárója

### A tagsági függvény értelmezése (gyakorlati megvalósításai)

Tagsági függvény a változó igazságtartalmát adja meg

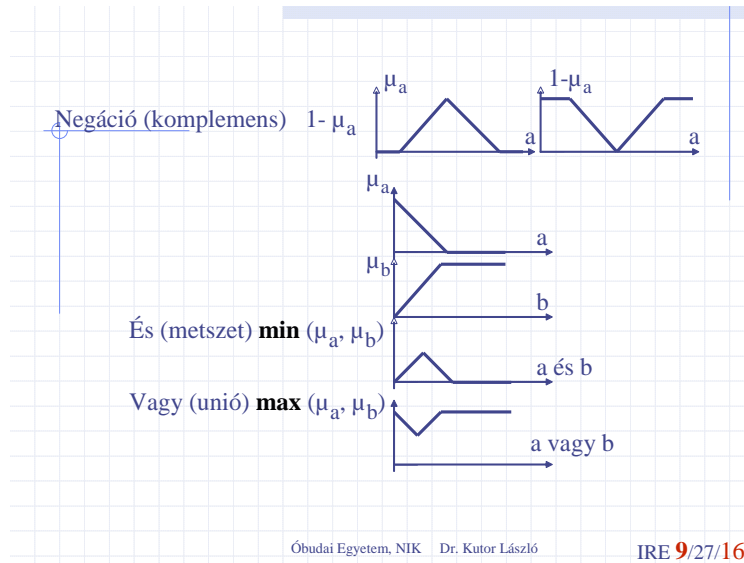


### A alapvető logikai függvények (és, vagy, komplement) értelmezése a Fuzzy logikában

Két fuzzy halmaz AND (metszet) művelete az a halmaz, amely a két argumentum halmaz közös elemeit tartalmazza, minden elemet véve a legkisebb előforduló beletartozási értéken. (MIN)

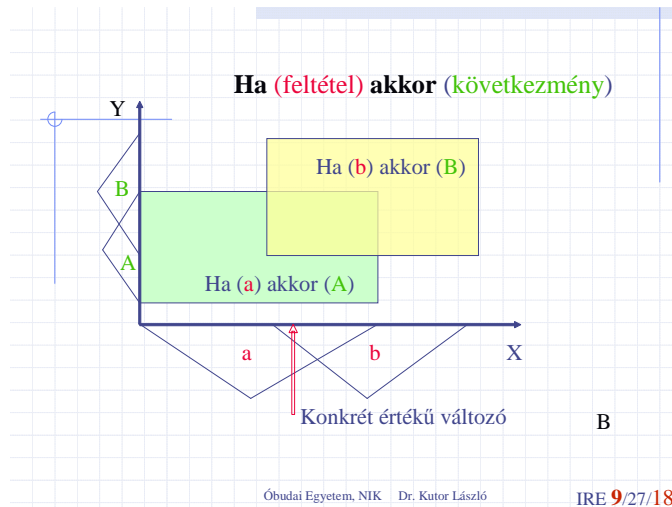
Két fuzzy halmaz OR (unió) művelete az a halmaz, ami minden előforduló elemet tartalmaz a lehető legnagyobb beletartozási értéken véve. (MAX)

Egy halmaz negáltján azt a halmazt értjük, mely tartalmazza az összes elemet, de az eredmény halmaz elemeinek tagsági értékeit kivonjuk 1-ből.

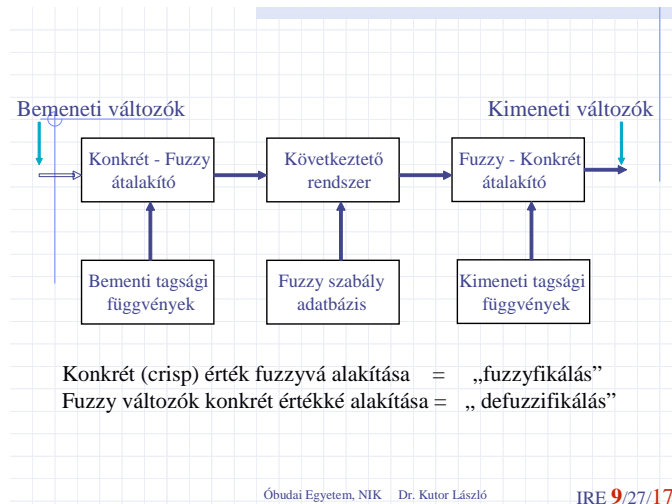


### A szabály értelmezése a Fuzzy logikában

Területként értelmezhető, ami átfogja a fuzzy változók által kijelölt tartományt



### A Fuzzy logikát alkalmazó szabályalapú rendszer blokkvázlata

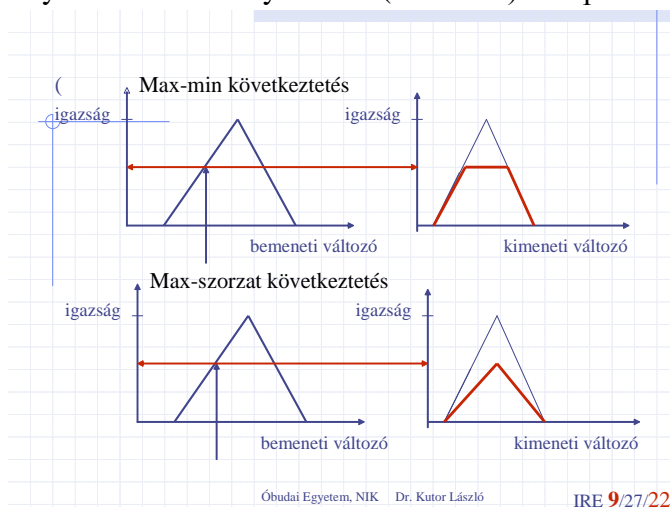


## A Fuzzy logika alkalmazásának menete

1. A bemeneti és kimeneti változók és tagsági függvényeinek meghatározása.
2. A Fuzzy szabályok létrehozása.
3. Következtető (válaszkiválasztó) mechanizmus kiválasztása.
4. Szimulátor segítségével a rendszermodell működésének ellenőrzése, „behangelása”.

## A Mamdani-féle következtetés értelmezése

A következtetés eredményeként keletkező fuzzy halmazt a bemenő adatok fuzzy halmaza és a szabálybázist leíró fuzzy reláció (max-min) kompozíciójaként állítja elő



## Miért van szükség válasz kiválasztó technikákra a Fuzzy logikában?

A változók gyakran átfedő tagsági függvényei miatt gyakran egyszerre több szabály is aktivizálódik. Technikai rendszerekben az egyértelmű válasz létrehozásához (fuzzy-crisp átalakításhoz) a számos egyidejűleg aktív szabályból létre kell hozni az egyértelmű kimenetet.

## A maximáló, a „singleton”, a központi valamint a súlyozott átlagoló módszer lényege

A maximáló válaszkiválasztó módszer szerint azt a válasz alkalmazzuk melynek a maximális az igazság értéke.

Singleton módszer szerint a tagsági függvényt egyetlen értékkel helyettesítjük.

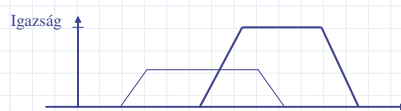
A központi módszernél a tagsági függvény középvértékét választjuk kimenetnek

Súlyozott átlagoló módszer szerint az aktivált szabályok kimeneteinek (tagsági függvényeinek) súlyozott átlagát vesszük ahol a súly = a tagsági függvény területével

## Válaszkiválasztó technikák 1.

Céljuk: A számos egyidejűleg aktivizált szabály kiértékelésével egy konkrét válasz kiszámítása

1. Maximáló módszer: Az aktivizált (egyidejűleg működő („tüzelő”) szabályok kimenő változói közül a legnagyobb hatást adót választjuk



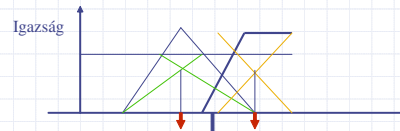
2. „singleton” módszer



## Válaszkiválasztó technikák 2.

Céljuk:: A számos egyidejűleg aktivizált szabály kiértékelésével egy konkrét válasz kiszámítása

### 3. Központ (centroid) módszer

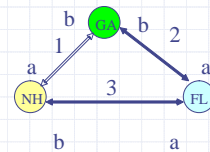


4. Súlyozott átlagoló módszer: Az aktivált szabályok kimeneteinek (tagsági függvényeinek) súlyozott átlagát vesszük

Súly = a tagsági függvény területe

$$k = (XaSa + XbSb) / 2$$

## A biológiai inspirációjú, ún. "lágyszámítási modellek" kapcsolatai



- 1 a.) a neurális hálózat tanítása (súlykeresés), topológia megkeresése  
b.) az egyed rátermettségének változtatása a tesztelés során
- 2 a.) Fuzzy változók tagsági függvényeinek meghatározása,  
Fuzzy szabályok keresése  
b.) Fuzzy kiértékelő módszerek alkalmazása az egyedek rátermettségének meghatározására
- 3 a.) a neurális hálózatok adaptív tulajdonságainak bevitel  
a Fuzzy logikát alkalmazó rendszerekbe.  
b.) szabályok automatikus feltárása tapasztalati adatokból

## A különböző modellek elhelyezkedése formalizált tudás dimenziójában

### A tudás dimenziója



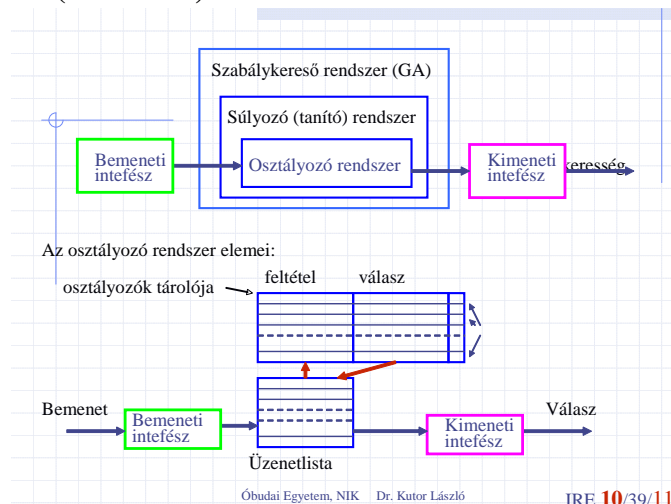
## Az elosztott mesterséges intelligencia rendszerekben megoldandó alapvető problémák?

- Hogyan írjuk le, bontjuk fel és rendeljük hozzá a problémákat a feladat megoldására szerveződő külön-külön intelligens alrendszerek között, valamint a részeredményeket hogyan szintetizáljuk?
- Hogyan biztosítsuk az alrendszerek kommunikációját és interakcióját?
- Milyen nyelvet, protokollt használjanak, és mikor kommunikáljanak?
- Hogyan biztosítsuk az alrendszerek koherens viselkedését, és hogyan kerüljük el az esetlegesen káros összeütközéseket?
- Hogyan jön létre az alrendszerek működésének ütemezése?
- Milyen technikai megoldásokat használunk az elosztott mesterséges intelligencia rendszerek tervezésére és építésére, optimalizálására?

## A beágyazott „ambiens” intelligens rendszer jellemzői

- A befogadó fizikai/kémiai/biológiai környezetükkel intenzív, valós idejű információs kapcsolatban álló,
- Autonóm működésű,
- Szolgáltatás-biztos (dependable),
- “láthatatlan” számítógépes rendszerek, melyek
- Lokálisan (általában) korlátozott,
- globálisan (általában) bőséges erőforrásokkal (idő, adat, tápellátás, memória, ...) rendelkeznek.

## A Holland-féle osztályozó (Classifier) rendszer főbb részei



Magja az ún. osztályozó rendszer, ami több részből áll:

1. A bemeneti interfész ami a környezet aktuális állapotát szabványos üzenetté alakítja át.
2. Az osztályozók tárolója: a rendszer információ feldolgozó eljárásait leíró szabályokat tartalmazza.
3. Az üzenet lista: egyrészt a bemeneti interfésztől, másrészt az aktivizált és engedélyezett szabályoktól érkező üzeneteket tartalmazza.
4. Kimeneti interfész: átalakítja az üzenetek egy részét olyan rendszer-válaszokká, melyek módosítják a környezet állapotát.

Az osztályozó mag működését egy inkrementális súlyozó-tanító rendszer a működés során lépésről-lépésre változtatja, javítja.

A rendszerben új szabályok keresését a genetikus algoritmus alkalmazása biztosítja. Erre azért van lehetőség, mert a szabályok felfoghatók egy egységes leírású elemekből álló populációnak, melyben az egyedi szabályok sikerességi mutatóval is rendelkeznek.

### A "Bucket Brigade" algoritmus célja és működése?

A Bucket Brigade nevű algoritmus a Holland-féle osztályozó rendszer tanuló algoritmus, ami elosztott, inkrementális tanulást biztosít. Feladata a működésben részt vevő szabályok sikerességi mutatójának lépésről-lépésre való módosítása.

Az osztályozó (szabály) felépítése:

**Ha (feltétel) akkor (következmény, hatás vagy üzenet), sikeresség(s(t))**

A szabály aktivizálásában szerepet játszó tényezők:

**R(c)** = feltétel megfelelés (Relevance), **S(c,t)** = korábbi sikeresség (Strength)

A szabályok az aktivizálásukért versenyeznek. A verseny a felajánlás **B(c)** (Bid) alapján dől el

$$B(c,t) = k * R(c) * S(c,t)$$

A győztes a felajánlást átadja annak a szabálynak, ahonnan az üzenet kapta.

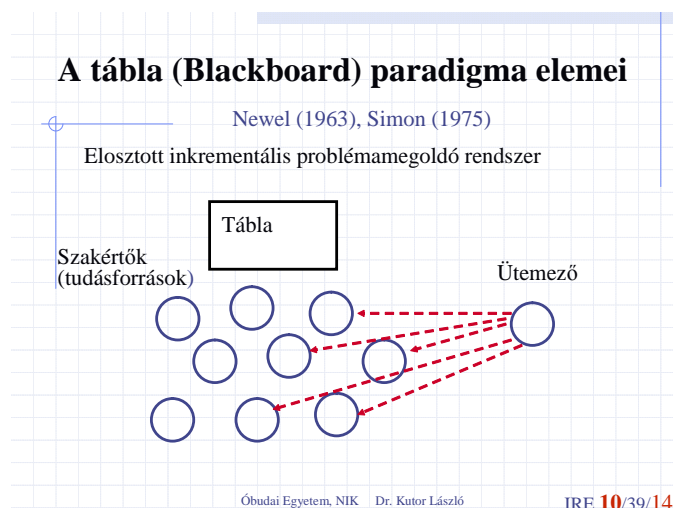
### A Holland féle osztályozó rendszer

A Holland féle osztályozó rendszer egy hierarchikus szabály alapú párhuzamos problémamegoldó rendszer. Főbb jellemzői:

- Párhuzamos rendszer.
- Üzenet-továbbításos rendszer.
- Szabály alapú tudásábrázolás.
- Adaptív, a tanulás az un. Bucket Brigade algoritmusra épül.
- Folyamatos szabálykereső mechanizmussal rendelkezik, mely a genetikus algoritmusokra épül.

### A tábla (blackboard) paradigma lényege, főbb komponensei

A tábla paradigma egy elosztott párhuzamos probléma-megoldó paradigma.



1. Szakértők, tudásforrások, melyek jellemzői:
  - független alrendszerek, melyek belső tudásábrázolása és következtető mechanizmusa rejtett,
2. Tábla, melyen keresztül minden kommunikáció lebonyolódhat. Úgy

Felfogható mint egy kommunikációs interfész, egy általánosan hozzáférhető adatbázis (kollektív emlékezet), valamint egy átmeneti tároló és trigger mechanizmus

3. Ütemező, ami mint felügyeli, ütemezi és irányítja a problémamegoldás menetét

figyelembe véve az egyes tudás források közreműködéséből eredő várható hasznot, és nyomon követi a problémamegoldás menetét

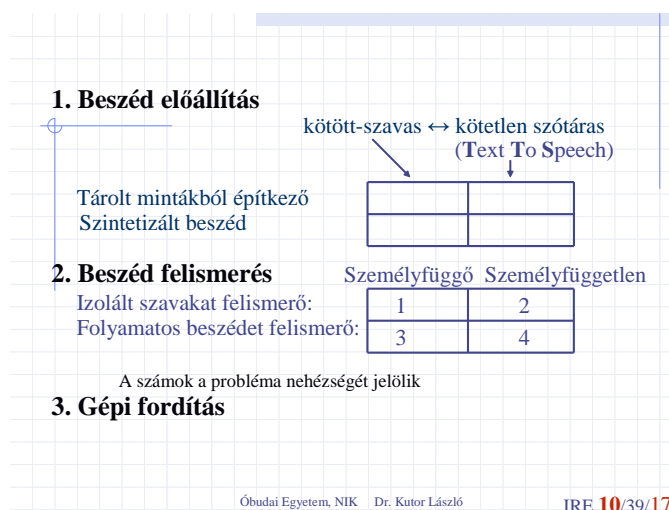
**A tábla- (blackboard) paradigma legfontosabb jellemzői**

- Inkrementális problémamegoldás
- A tudás források (szakértők) függetlensége, moduláris rendszer. Újabb szakértők adhatók a rendszerhez a korábbiak módosítása nélkül. A gyengén szereplő szakértők munkája javítható, vagy eltávolíthatók a rendszerből.
- Különböző probléma-megoldó technikák egyidejűleg alkalmazhatók. Minden szakértő fekete doboznak tekinthető, csak a kommunikációs nyelvük közös
- Flexibilis tudásábrázolás
- Nincsenek korlátozások, hogy milyen információk kerülhetnek a táblára
- Közös az interakciós nyelv, melynek szintakszisa és a szemantikája előre meghatározott
- Eseményalapú aktiváció, célvezérelt aktivitás. Minden szakértő keresi a lehetőséget, hogy közreműködjön a feladat megoldásában
- A vezérlés szükségessége: ki írhat a táblára, és mi a közreműködés ára.

**A képfeldolgozás klasszikus területei, megoldandó feladatai és jellemzői**

- Alapvető képfeldolgozás (kép-kép leképezés): digitális kép létrehozása, a leképezési hibák kijavítása, jellemző tulajdonságok kiemelése
- Alakfelismerés (jellemzők meghatározása, azonosítása): a képet jellemző sajátosságok felismerése a kép a képen előforduló alakzatok azonosítása
- Képfelismerés (leírás, értelmezés, azonosítás): a képen rögzített objektumok Felismerése adatbázis alapján

**A nyelvtechnológia klasszikus területei, megoldandó feladatai és jellemzői**



**Robotok generációi, jellemzői**

- *Első generáció:* kizárólag vezérléssel működtethetők, a környezet változásait nem érzékelik.



- *Második generáció:* környezetüket szenzorokkal vizsgálják, az így szerzett és a saját működésükről nyert információk alapján a számítógép bármikor képes módosítani a robot mozgását, például kikerüli a váratlanul útjába került akadályokat. Feladataikat magas szintű programnyelven határozzák meg.
- *Harmadik generáció:* alkalmazkodnak a környezet változásaihoz, alakokat és helyzeteket ismernek fel, hanggal is vezérelhetők, esetenként hanggal tudnak válaszolni, önálló döntéseket hoznak, bonyolult feladatokat oldanak meg, tanuló mechanizmusokkal is rendelkeznek.

### Az inkrementális-, alulról építkező robotépítés célja, elvárásai és javasolt módja (Brooks)

Cél: Olyan autonóm és intelligensnek tekinthető lény létrehozása, amely az emberekkel együtt létezik (él?).

Elvárások az autonóm és intelligens viselkedő lényekkel szemben:

- Megfelelően és időben reagáljon a dinamikus változó környezeti hatásokra.
- Robusztus legyen: a környezet kisebb nagyobb változása ne okozhassa a viselkedés összezavarodását.
- Célokkal kell rendelkeznie. „szerepe van a földön”

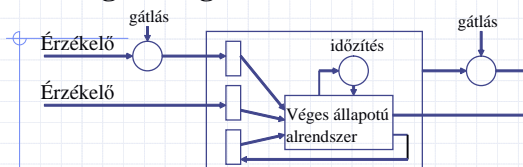
A megvalósítás javasolt módja:

- Inkrementális építkezés
- Minden egyes részfunkció megvalósításánál biztosítjuk a rendszer teljes működését
- Minden alrendszer önálló érzékelő és vonatkozó képességgel rendelkezzen
- Az alrendszerek üzenetekkel tartják egymással a kapcsolatot, egymást serkentik vagy gátolják.

### A Brooks féle magába foglaló architektúrát jellemzői, a tipikus alrendszer vázlata

- A részfunkciók önálló megvalósítása, az alrendszereknek nincs szüksége a felettük levő rendszerek kontrolljára
- Minden rétegnek önálló szerepe van
- Az összetett rendszer építése inkrementálisan, lépésről lépésre történik, a magasabb rétegek építik az alacsonyabb rétegeket
- A céloknak nincs központi reprezentációja, amelyek közül valamilyen központi eljárás választana. Minden alrendszer végzi a feladatát
- Az alrendszerek tesztelése mindig a valóságos helyzetben történik. Így elkerülhető, hogy csak leegyszerűsített un. játék problémákkal foglalkozzon
- A rétegek az elnyomás és a tiltás-gátló mechanizmusokon keresztül működnek együtt

#### A magába foglaló architektúra vázlata



A véges állapotú alrendszer részei:

- ♦ regiszterek (üzenetek fogadására, belső állapotok tárolására)
- ♦ véges állapotú gép egyszerű információ feldolgozó
- ♦ kapacitással belső időzítők

Az alrendszerek jellemzői:

- ♦ rögzített hosszúságú üzeneteket fogad és küld
- ♦ aszinkron

## Az ágens paradigma értelmezése, tipikus ágens feladatok, és alkalmazások

Az ágens paradigma a feladatmegoldás egy új megközelítése. Az objektum orientált paradigmához képest itt az egységek még több mindent foglalnak magukban. Az ágensek esemény vagy célvezérelt autonóm programok, amelyek valaki érdekében cselekszenek. Szokás őket „soft botok”-nak is nevezni.

Tipikus ágens feladatok:

- A bejövő információk szűrése, megválaszolása
- Döntéstámogatás és szakértői (különböző adatbázisok alapján)
- Ismétlődő feladatok elvégzése (pl: rendszergazdai munkák, bevásárló ágens, kereskedő ágens, interfész ágens)
- Titkári feladatok (programszervezés)

### Ágensek tulajdonságai

◆ beágyazottság	Antropomorf tulajdonságok
◆ reaktivitás	(mentális állapotok):
◆ tudás	◆ racionalitás
◆ helyzetfüggőség	◆ szándék
◆ autonómia	◆ megbízhatóság
◆ kezdeményező képesség	◆ igazmondás
◆ célvezérelt viselkedés	◆ vélekedés
◆ tanuló képesség, adaptivitás	◆ elkötelezettség
◆ időbeni állandóság	◆ érzelmek
◆ kommunikációs képesség	◆ személyiség

### Ágensekkel kapcsolatos feladatok, problémák

- ◆ Célok illetve a felhasználói igények tárolása
- ◆ A feladat végrehajtásához szükséges ismeretek tárolása, kezelése
- ◆ A cél eléréséhez szükséges erőforrások elérhetőségének leírása
- ◆ Költség és időhatárok kezelése

Megoldandó problémák

- ◆ Platform-független nyelv (JAVA, Applescript, TeleScript)
- ◆ számlázás (adók)
- ◆ információ védelem
- ◆ információforrás megkeresése
- ◆ kommunikáció
- ◆ felelősség