AZ MI FOGALMA

mesterséges intelligencia – MI *(artificial intelligence - AI)*

sokan sokfélét értenek alatta

Nem egy rész-területe az informatikának, hanem egy szemléletmód, amely az informatika fejlődését szolgálja: olyan problémákra keres számítógépes megoldásokat, amelyek megoldásában az ember jobbnak tűnik.

Erős MI
Cél: az emberi gondolkodás számítógéppel történő reprodukálása.

MI szkeptikusok
A számítógép soha nem lesz okosabb az embernél.

Gyenge MI
Cél: Azon elméletek és módszerek kutatása, fejlesztése, rendszerezése, amelyekkel az emberi intelligencia számára is érdekes és nehéz problémákra adhatunk számítógépes megoldásokat.

MI története

módszerek és célok specializálódása

<u>Projektek</u>:

kétszemélyes játékok (sakk)

beszélgető program (ELIZA,1966)

Romantikus kor

mindenféle problémát általános eszközökkel

1956 1960 1970 1980 1990 2000 2010

ELIZA

Minta-válasz párok:

Úgy érzem, hogy ön mostanában engem un.

<a> ön <b> engem <c>.

1. Miért gondolja, hogy ön <a> én <b> <c>?
2. Tegyük fel, hogy én <b> önt <c>. Mit változtat ez a dolgokon?

Ismétlés felismerése:

„Miért ismételgeti ugyanazt újra és újra?”

Folytatás:

Igen, értem. Kérem folytassa. Ez nagyon érdekes.
Még miről szeretne beszélgetni?

módszerek és célok specializálódása

MI története

<u>Projektek</u>:

kétszemélyes játékok (sakk)

beszélgető program (ELIZA,1966)

<u>Módszerek</u>:

GPS, rezolúció (1966)

Lisp (1958)

mesterséges neuronhálók

evolúciós algoritmusok

Romantikus kor

mindenféle problémát általános eszközökkel

1956 1960 1970 1980 1990 2000 2010

MI története

módszerek és célok specializálódása

Klasszikus kor

speciális problémákat
speciális módszerekkel

Romantikus kor

mindenféle problémát
általános eszközökkel

Projektek:
- SHRDLU (1972),
- BACON, AM
- DENDRAL (1969-78),
- MYCIN(1976)

Módszerek:
- heurisztikus keresés,
- tudás reprezentáció
- Prolog

1956 1960 1970 1980 1990 2000 2010

MI története

módszerek és célok specializálódása

Ipari kor

szakértő rendszerek
tudásalapú rendszerek

Klasszikus kor

speciális problémákat
speciális módszerekkel

Romantikus kor

mindenféle problémát
általános eszközökkel

Projektek: XCON(1982),
PROSPECTOR (1979)

Módszerek:
- shell-ek (IDE)
- tudás-alapú technológia
- nem-klasszikus következtetés
- bizonytalanság kezelés

1956 1960 1970 1980 1990 2000 2010

MI története

módszerek és célok specializálódása

Romantikus kor

mindenféle problémát
általános eszközökkel

Klasszikus kor

speciális problémákat
speciális módszerekkel

Ipari kor

szakértő rendszerek
tudásalapú rendszerek

MI tél, majd reneszánsz

gépi tanulás,
hibrid technológiák

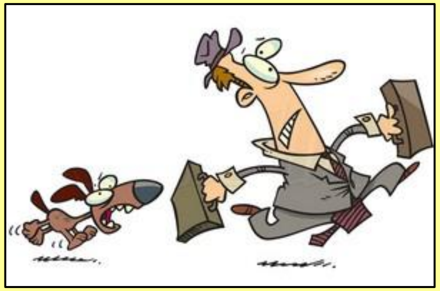
Projektek:
- Deep Blue (1997)
- IBM Watson (2011)
- robotika

Aktuális területek:
- mély hálók
- felügyelet nélküli tanulás
- adattudomány

1956 1960 1970 1980 1990 2000 2010

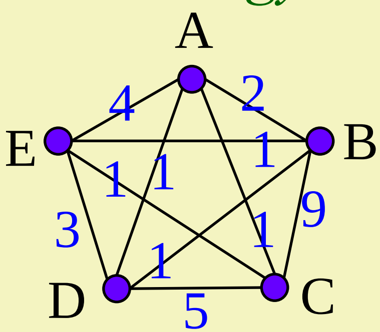
Miről ismerhető fel egy szoftverben az MI?

- Megoldandó feladat: nehéz
- Szoftver viselkedése
- Felhasznált technológiák



Utazó ügynök problémája

Adott n város a közöttük vezető utak költségeivel. Melyik a legolcsóbb olyan útvonal, amely az A városból indulva mindegyik várost egyszer érintve visszatér az A városba?



n	(n-1)!
5	24
50	$6\cdot10^{62}$

lehetséges utak:

ABCDEA

ACBDEA ABDCEA ABCEDA

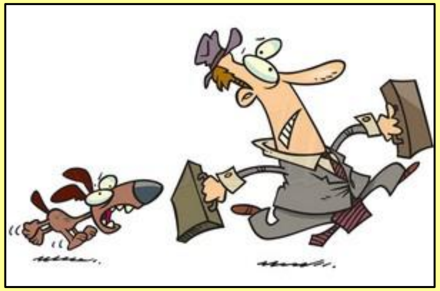
ABDECA

ADBECA …

problématér

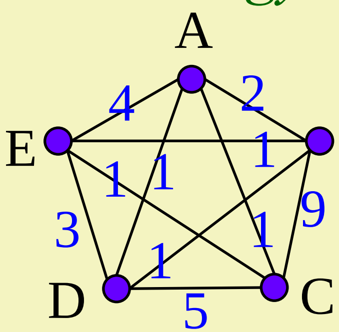
Miről ismerhető fel egy szoftverben az MI?

- Megoldandó feladat: nehéz
  - A feladat problématere hatalmas,
  - szisztematikus keresés helyett intuícióra, kreativitásra (azaz heurisztikára) van szükségünk ahhoz, hogy elkerüljük a kombinatorikus robbanást.
- Szoftver viselkedése
- Felhasznált technológiák

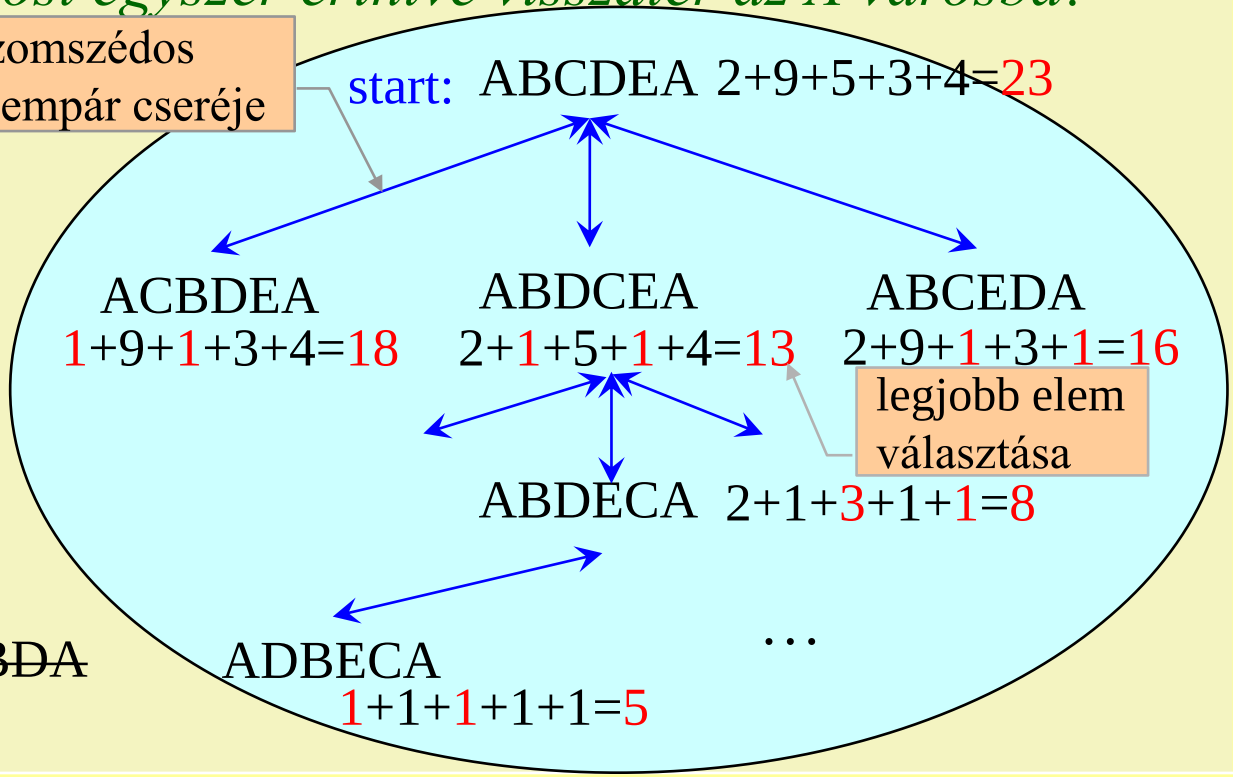


Utazó ügynök problémája

Adott n város a közöttük vezető utak költségeivel. Melyik a legolcsóbb olyan útvonal, amely az A városból indulva mindegyik várost egyszer érintve visszatér az A városba?



szomszédos elempár cseréje



legjobb elem választása

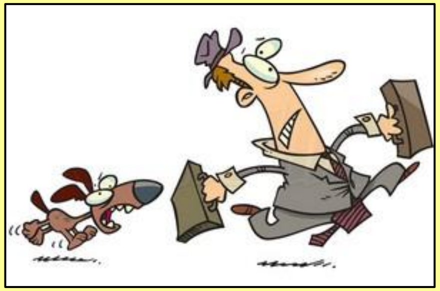
~~AEDCBA~~

~~AEDBCA~~

~~ACEDBA~~

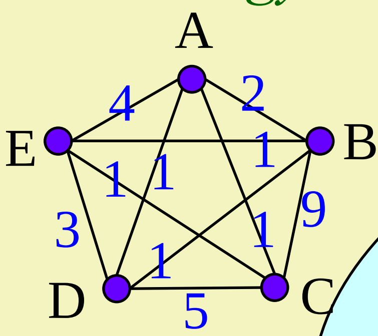
~~ACEBDA~~

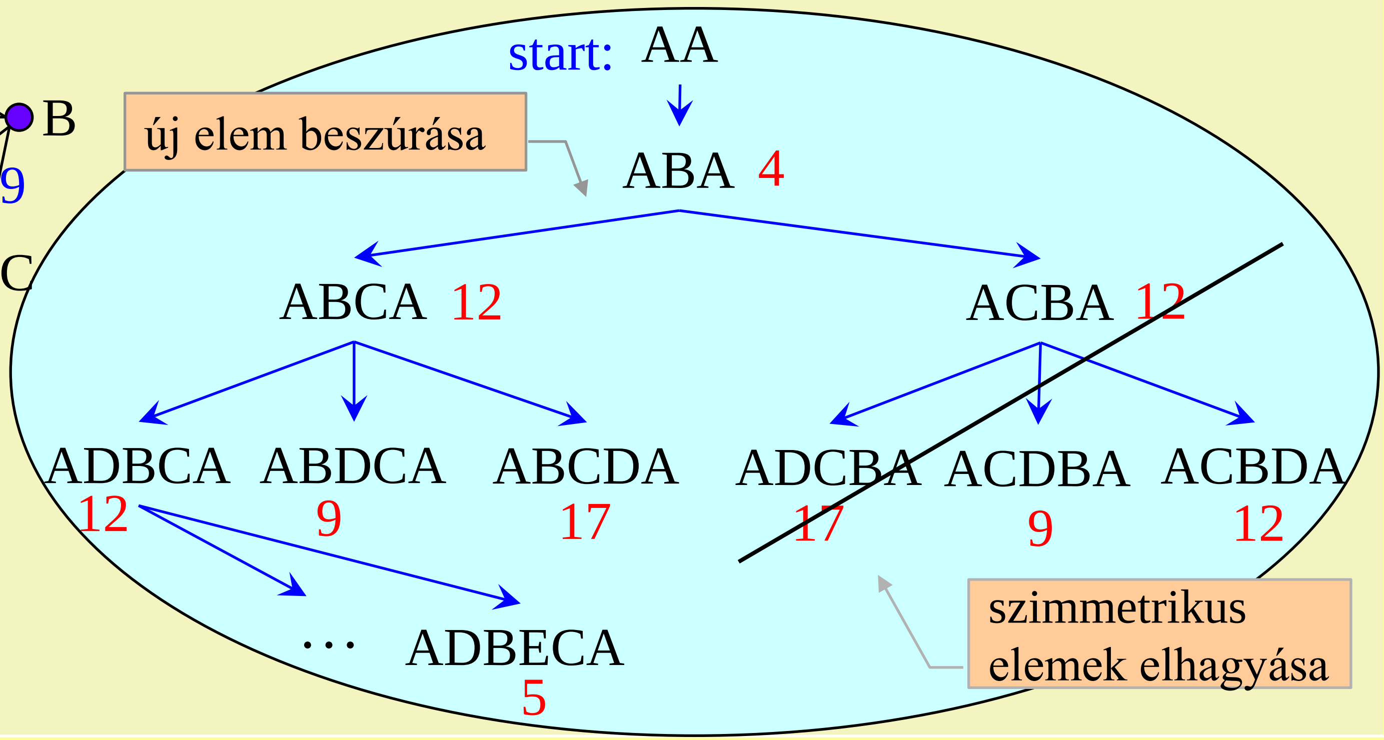
felesleges elemek elhagyása



Utazó ügynök problémája

Adott n város a közöttük vezető utak költségeivel. Melyik a legolcsóbb olyan útvonal, amely az A városból indulva mindegyik várost egyszer érintve visszatér az A városba?





Miről ismerhető fel egy szoftverben az MI?

- Megoldandó feladat: nehéz
  - A feladat problématere hatalmas,
  - szisztematikus keresés helyett intuícióra, kreativitásra (azaz heurisztikára) van szükségünk ahhoz, hogy elkerüljük a kombinatorikus robbanást.
- Szoftver viselkedése: intelligens
  - Turing teszt
- Felhasznált technológiák

Miről ismerhető fel egy szoftverben az MI?

- Megoldandó feladat: nehéz
  - A feladat problématere hatalmas,
  - szisztematikus keresés helyett intuícióra, kreativitásra (azaz heurisztikára) van szükségünk ahhoz, hogy elkerüljük a kombinatorikus robbanást.
- Szoftver viselkedése: intelligens
  - Turing teszt vs. kínai szoba elmélet
  - általános mesterséges intelligencia
- Felhasznált technológiák: sajátosak
  - speciális reprezentáció a feladat modellezéséhez
  - heurisztikával megerősített hatékony algoritmusok
  - gépi tanulás módszerei

Intelligens szoftver jellemzői
- megszerzett ismeret tárolása
- automatikus következtetés
- tanulás
- term. nyelvű kommunikáció

\+ gépi látás, gépi cselekvés

modellezés és keresés

gépi tanulás

MODELLEZÉS & KERESÉS

modellezés — keresés

feladat → útkeresési probléma → megoldás

gráfreprezentáció

Állapottér modell
Probléma redukció
Probléma dekompozíció
Korlátprogramozási modell
Kétszemélyes játék modellje
Logikai reprezentációk
Valószínűségi háló

Lokális keresések
Visszalépéses keresés
Gráfkeresések
Evolúciós algoritmus
Játékfa kiértékelő módszerek
Logikai következtetések
Bizonytalanság kezelés

Mire kell a modellezésnek fókuszálni

- ***Problématér elemei***: probléma lehetséges válaszai
- ***Cél***: egy helyes válasz (megoldás) megtalálása
- ***Keresést segítő ötletek*** (heurisztikák):
  - Problématér hasznos elemeinek elválasztása a haszontalanoktól.
  - Kiinduló elem kijelölése.
  - Az elemek szomszédsági kapcsolatainak kijelölése, hogy a probléma tér elemeinek szisztematikus bejárását segítsük.
  - Adott pillanatban elérhető elemek rangsorolása.

Útkeresési probléma

- Útkeresési probléma az, amelynek megoldása megfeleltethető egy élsúlyozott irányított gráfbeli
  - csúcsnak (célcsúcs), vagy még inkább
  - útnak (startcsúcsból célcsúcsba, esetleg a legolcsóbb)

Számos olyan modellező módszert ismerünk, amely a kitűzött feladatot útkeresési problémává fogalmazza át.

- Ez a gráf (δ-gráf) lehet végtelen nagy, de
  - csúcsainak kifoka véges, és
  - élei súlyának (költségének) van egy konstans globális pozitív alsó korlátja (δ).

Gráf fogalmak 1.

- csúcsok, irányított élek — $N,\ A \subseteq N \times N$ (végtelen számosság)
- él *n*-ből *m*-be — $(n,m) \in A\ (n,m \in N)$
- *n* utódai — $\Gamma(n) = \{m \in N \mid (n,m) \in A\}$
- *n* szülei — $\pi(n) \in \Pi(n) = \{m \in N \mid (m,n) \in A\}$
- irányított gráf — $R=(N,A)$
- véges sok kivezető él — $|\Gamma(n)| < \infty \quad (\forall n \in N)$
- élköltség — $c: A \to \mathbb{R}$
- δ-tulajdonság ($\delta \in \mathbb{R}^+$) — $c(n,m) \geq \delta > 0 \quad (\forall (n,m) \in A)$
- δ-gráf — δ-tulajdonságú, véges sok kivezető élű, élsúlyozott irányított gráf

Gráf fogalmak 2.

- irányított út — $\alpha = (n,n_1),(n_1,n_2),\dots,(n_{k-1},m)$
  $= \langle n,n_1,n_2,\dots,n_{k-1},m\rangle$

  $n \longrightarrow^{\alpha} m,\ n \longrightarrow m,\ n \longrightarrow M$ ($M \subseteq N$)

  $\{n \longrightarrow m\},\ \{n \longrightarrow M\}$ ($M \subseteq N$)

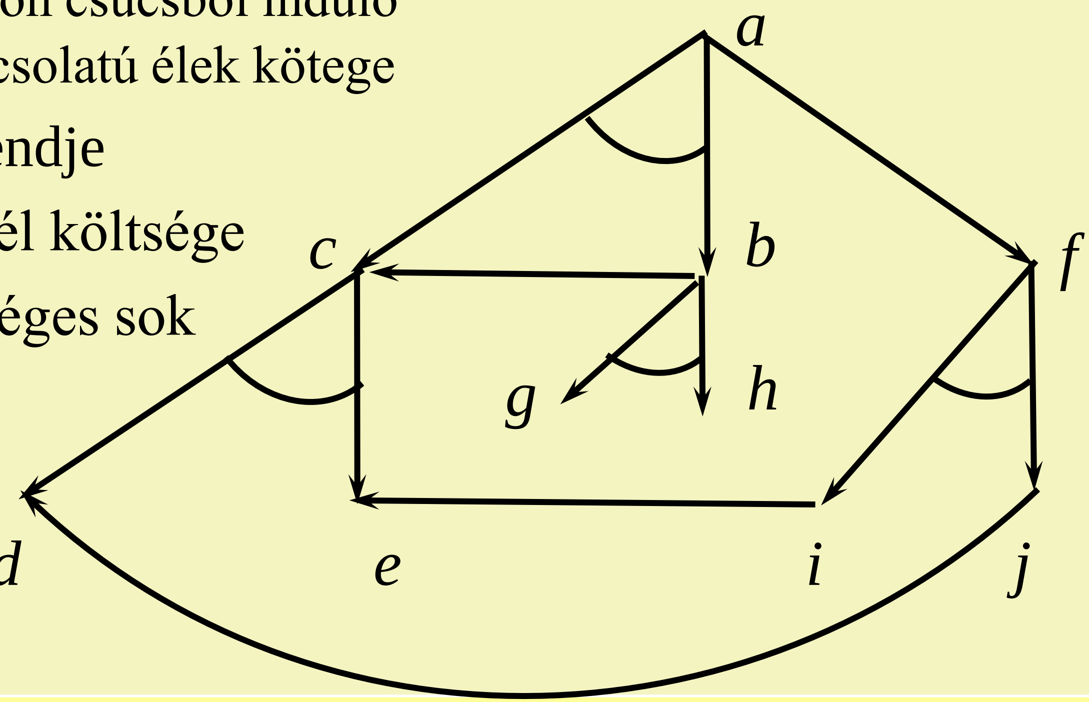
  δ-gráfokban ez végtelen sok út esetén is értelmes.

  Értéke ∞, ha nincs egy út se.

- út hossza — az út éleinek száma: $|\alpha|$
- út költsége — $c(\alpha)=c^{\alpha}(n,m) := \Sigma_{i=1..k}\ c(n_{i-1},n_i)$
  ha $\alpha = \langle n=n_0,n_1,n_2,\dots,n_{k-1},m=n_k\rangle$
- opt. költség — $c^*(n,m) := \min_{\alpha\in\{n\longrightarrow m\}} c^{\alpha}(n,m)$
  $c^*(n,M) := \min_{\alpha\in\{n\longrightarrow M\}} c^{\alpha}(n,m)$
- opt. költségű út — $n \longrightarrow^{*} m := \min_c\{\alpha \mid \alpha \in \{n \longrightarrow m\}\}$
  $n \longrightarrow^{*} M := \min_c\{\alpha \mid \alpha \in \{n \longrightarrow M\}\}$

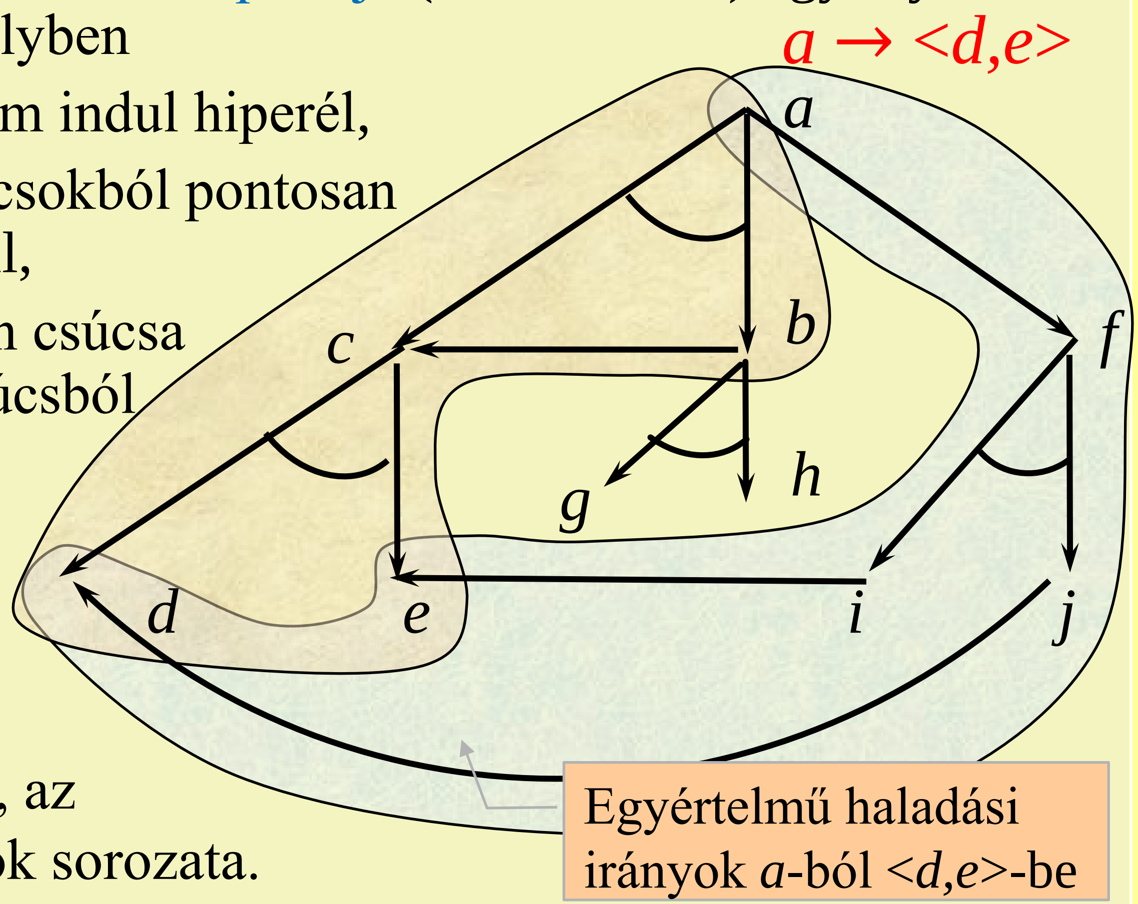
ÉS/VAGY gráfok

- $R=(N,A)$ élsúlyozott irányított hipergráf, ahol
  - N a csúcsok halmaza
  - $A \subseteq \{ (n,M) \in N \times N^{+} \mid 0 \neq |M| < \infty \}$ a hiperélek halmaza
    hiperél ~ ugyanazon csúcsból induló
    ÉS kapcsolatú élek kötege
  - $|M|$ a hiperél rendje
  - $c(n,M)$ az (n,M) él költsége
- Egy csúcsból csak véges sok hiperél indulhat.
- $0<\delta\leq c(n,M)$



Az n csúcsból az M csúcs-*sorozatba vezető irányított hiperút fogalma*

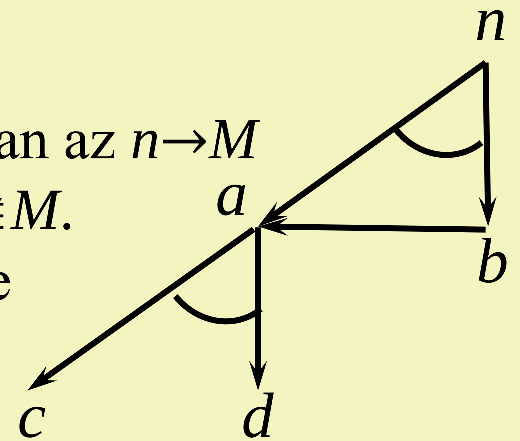
- Egy ÉS/VAGY gráf $n^{\alpha} \to M$ hiperútja ($n \in N$, $M \in N^{+}$) egy olyan véges részgráf, amelyben
  - *M* csúcsaiból nem indul hiperél,
  - *M*-en kívüli csúcsokból pontosan egy hiperél indul,
  - a hiperút minden csúcsa elérhető az *n* csúcsból egy közönséges irányított úton.



- A megoldás-gráf egy $s \to M$ hiperút, ahol *s* a startcsúcs, az *M* pedig célcsúcsok sorozata.

A hiperút bejárása

- Az $n \rightarrow M$ hiperút bejárását a hiperéleinek adott sorrendű felsorolásával kapjuk, amelyet a hiperút csúcsaiból képzett csúcs-sorozatok felsorolásával is megadhatunk :
  - első sorozat: $<n>$
  - C sorozatot a $C^{k \leftarrow K}$ sorozat követi, ha van az $n \rightarrow M$ hiperútban (k,K) hiperél, és $k \in C$, de $k \notin M$. $C^{k \leftarrow K}$ úgy kapjuk, hogy C-ben a k helyére mindenhol K-t írunk.



- Így egy hiperutat közönséges irányított útként foghatunk fel igaz többféleképpen is, mert több bejárása is lehet:

$$<n> \;\rightarrow <a,b> \;\rightarrow <a,a> \;\rightarrow < c,d,c,d >$$

$$<n> \;\rightarrow <a,b> \;\rightarrow <c,d,b> \;\rightarrow <c,d,a> \;\rightarrow < c,d,c,d >$$

Gráfreprezentáció fogalma

- Minden útkeresési probléma rendelkezik egy (a probléma modellezéséből származó) gráfreprezentációval, ami egy (R,s,T) hármas, amelyben
  - $R=(N,A,c)$ a reprezentációs gráf (δ- vagy ÉS/VAGY gráf)
  - az $s\in N$ startcsúcs,
  - a $T\subseteq N$ halmazbeli célcsúcsok.
- és a probléma megoldása:
  - t cél vagy $<t_1, \dots , t_m>$ célcsúcs-sorozat megtalálása ($t, t_1, \dots , t_m \in T$), vagy
  - $s\longrightarrow t$ vagy $s\longrightarrow <t_1, \dots , t_m>$ esetleg egy optimális $s\longrightarrow^* T$ út megtalálása

Útkeresés δ-gráfban

- Egy útkeresési probléma megoldásához a reprezentációs gráfjának nagy mérete miatt speciális (nem-determinisztikus, heurisztikus) útkereső algoritmusra van szükség, amely
  - a startcsúcsból indul (kezdeti aktuális csúcs);
  - minden lépésben nem-determinisztikus módon új aktuális csúcso(ka)t választ a korábbi aktuális csúcs(ok) segítségével (gyakran azok gyerekei közül);
  - tárolja a már feltárt reprezentációs gráf egy részét;
  - megáll, ha célcsúcsot talál vagy nyilvánvalóvá válik, hogy erre semmi esélye.

Útkeresés ÉS/VAGY gráfban

- ÉS/VAGY gráfbeli megoldás-gráf keresése visszavezethető egy δ-gráfban történő útkeresésre.
- A startcsúcsból induló hiperutakat (köztük a megoldás-gráfokat is) a bejárásukkal (közönséges irányított utakkal) ábrázolhatjuk, amelyek egy δ-gráfot(!) határoznak meg. A δ-gráf
  - csúcsai az eredeti ÉS/VAGY gráf csúcsainak sorozatai
  - startcsúcsa az ÉS/VAGY gráf startcsúcsából álló sorozat
  - célcsúcsai az ÉS/VAGY gráf célcsúcsaiból álló sorozatok
- Az így nyert δ-gráf megoldási útjai az eredeti ÉS/VAGY gráfbeli megoldás-gráfokat reprezentálják. Ezért egy ÉS/VAGY gráfban a megoldás-gráf megkeresése a neki megfeleltetett δ-gráfban történő megoldási út megkeresésével helyettesíthető.