



Mesterséges
Intelligencia

Csató Lehel

Mesterséges Intelligencia

Csató Lehel

Matematika-Informatika Tanszék
Babeş–Bolyai Tudományegyetem, Kolozsvár

2007/2008



Az Előadások Témái

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési

Feladatok

- Bevezető: mi a *mesterséges* intelligencia ...
- „Tudás”-reprezentáció
- Gráfkeresési stratégiák
- Szemantikus hálók / Keretrendszerek
- Játékok modellezése
- Bizonytalanság kezelése
- Grafikus modellek
- Tanuló rendszerek
- Szimulált kifűtés, Genetikus algoritmusok
- Neurális hálók
- Gépi tanulás
- Nemparametrikus módszerek



Mesterséges
Intelligencia

4

Csató Lehel

Algoritmusok

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési

Feladatok

Előadások honlapja

<http://www.cs.ubbcluj.ro/~csatol/mestint>

Vizsga

Szóbeli (60%) + Gyakorlat (40%)
(v) Előadás (60%)

Laborgyakorlatok:

- | | | |
|---|---|----------|
| 1 | Clean vagy Prolog - dedukciós algoritmus | 30% |
| 2 | C / C++ / C# / ... - genetikus algoritmus | 10% vagy |
| 3 | Matlab - Neurális hálózatok vagy SVM | 10% |

A számítástudomány és a molekuláris biológia között²

David Haussler & Judea Pearl

Kifejlesztették az emberi genomot feltérképező programokat. A lehetőséget a számítógéptechnológia és a biokémia fejlődése biztosította. A genom biológiai összetevőinek felderítését és elemzését a tudós által kidolgozott

valószínűségi megközelítés alapozta meg.

Az emberi génállomány mintegy **hárommilliárd alappárt** képez: a kettős spirál-alakú DNS négy alap-nukleotidból (**A, C, G, T**) épül fel; Minden egyes nukleotid része egy párnak. A mennyiség nagyon nagy, a munka csak számítógépes módszerekkel végezhető el.





Bioinformatika

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési

Feladatok

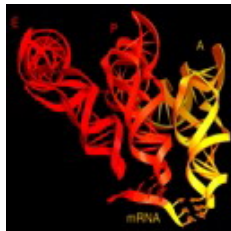
2001. júliusában közzétették a módszer vázlatait, majd az emberi és egyéb organizmusok (egér, patkány, stb.) génszekvenciáit elemző, jegyzetekkel ellátott interaktív webalapú keresőket fejlesztettek.

Tudományos fórumot teremtettek, míg programjaikat gyakran használják különböző biomedikális kutatásoknál, kísérleteknél.

A gének evolúciója

A CBSE kutatásai az interdiszciplináris megközelítés jegyében folynak. Biológia, információs és nanotechnológia fúziójára, minél kisebb szerkezetek létrehozására törekednek.

Az emberi genom evolúciójának jobb megértése az egyik fő irány: a cél érdekében permanensen fejlesztik az új statisztikai és algoritmikus módszereket.

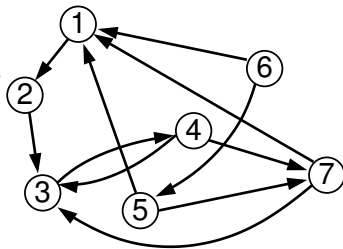


Feladatok:

- egy megoldás megtalálása
- minimális út keresése

Módszerek:

- Heurisztika
- Visszalépés





(* – Futó: Mesterséges Intelligencia, 73.o)

Visszalépés – backtracking – hátránya, hogy **nem** találja meg az optimális utat.

Gráfkereső algoritmus:

- a startcsúcsból indul S
- feltárja a reprezentációs gráfot G
 - 1 kiválaszt egy csúcsot, melynek utódai $n \in NYILT$
nem ismertek,
 - 2 kiterjeszti a választott csúcsot $G \leftarrow G \cup \Gamma(n)$
 $NYILT \leftarrow NYILT \setminus \{n\}$
 $NYILT \leftarrow NYILT \cup \Gamma(n)$
- addig keres, amíg egy célcsúcsot nem talál és van kiterjeszthető csúcs.



Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési

Feladatok

Kommutatív rendszer – a kiterjesztések bármilyen sorrendben végrehajthatók.



- A vezérlési stratégia nem informatív,
- Másodlagos stratégia $\Rightarrow$ továbblépés.

Módosítás:

$$n \leftarrow \underset{m \in NYILT}{\operatorname{argmin}} f(m)$$

ahol $f : NYILT \rightarrow \mathbb{R}$ kiértékelő függvény, amely

- egy csúcs „jósága”,
- pl. az s -ből m -be vivő legkisebb út hossza.
- $\Rightarrow$ dinamikus függvény. (felületes def.)



Bevezetjük:

- kitüntetett **szülő csúcsot** (parent)
(az **s**-ből **egy** utat specifikál).

$$p : G \rightarrow G$$
$$p(s) = \text{nil}$$

- **költségfüggvényt:**
 $g(m)$ az **s**-ből **m**-be vivő út költsége.

$$g : G \rightarrow \mathbb{R}$$

Az **n** csúcs minden **k** utód-csúcsára $\forall k \in \Gamma(n)$:

- Ha **k** új csúcs, vagy

$$k \notin G$$

- Ha nem új **és** $g(k) > g(n) + c(n, k)$, akkor

$$k \in G$$

$$p(k) \leftarrow n$$

$$g(k) \leftarrow g(n) + c(n, k)$$



Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési

Feladatok

Probléma:

- ha k zárt,
- korábban megkerestük utódjait
- és rövidebb utat találtunk,



- a g, p függvények nem helyesek a k utódain;
- a feszítőfa nem optimális.

Megoldások:

- 1 k összes leszármazottját újraértékeljük;
- 2 a k csúcsot visszatesszük a $NYILT$ halmazba.

Hátrány:

- Nagyobb futási idő;
 - p nem mindig optimális.
- 3 Olyan f választása, mely garantálja, hogy **nem lesz ilyen k .**



Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési

Feladatok

Probléma:

- ha k zárt,
- korábban megkerestük utódjait
- és rövidebb utat találtunk,



- a g, p függvények nem helyesek a k utódain;
- a feszítőfa nem optimális.

Megoldások:

- 1 k összes leszármazottját újraértékeljük;
- 2 a k csúcsot visszatesszük a $NYILT$ halmazba.

Hátrány:

- Nagyobb futási idő;
- p nem mindig optimális.

- 3 Olyan f választása, mely garantálja, hogy nem lesz ilyen k .



Algoritmus

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési

Feladatok

- $G \leftarrow \{s\}, NYILT \leftarrow \{s\}, g(s) \leftarrow 0, p(s) \leftarrow nil.$
- While **nem ures** ($NYILT$),
 - $n \leftarrow \arg \min_{m \in NYILT} f(m)$
 - If **cél**(n) then **kilép**.
 - $NYILT \leftarrow NYILT \setminus \{n\}$
 - For $\forall k \in \Gamma(n)$
 - If $k \notin G$ or $g(k) > g(n) + c(n, k)$
 - $p(k) \leftarrow n$
 - $g(k) \leftarrow g(n) + c(n, k)$
 - $NYILT \leftarrow NYILT \cup \{k\}$
 - endfor
 - $G \leftarrow G \cup \Gamma(n)$
- endwhile



Algoritmus tulajdonságai

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus
Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési
Feladatok

Tulajdonságok

Az általános gráfkereső algoritmus

- egy csúcsot véges sokszor terjeszt ki;
- véges gráfban mindig terminál;
- mindegyik $n \in NYILT$ csúcs kiterjesztése előtt $\forall s \rightarrow n$ van m csúcs az optimális úton, mely
 - 1 $m \in NYILT$,
 - 2 $g(m) = g^*(m)$,
 - 3 minden m -et előző csúcs az úton **zárt**;
- Egy véges gráfban, ha létezik megoldás, akkor az algoritmus egy célcsúcs megtalálásával terminál.
- Csökkenő kiértékelőfüggvény használata mellett optimális és konzisztens a feszítőfa.

Bizonyítás



Nevezetes gráfkereső algoritmusok

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélyégi

Szélességi

Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési

Feladatok

Futó: Mesterséges Intelligencia – pp. 83

- Nem-informált gráfkeresések

- 1 Mélységi keresés
- 2 Szélességi keresés
- 3 Egyenletes keresés

- Heurisztikus keresések

- 1 Előretekintő keresés
- 2 A algoritmus
- 3 A* algoritmus
- 4 A^c algoritmus



Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus
Általános algoritmus

Példa

Mélységi

Szélességi

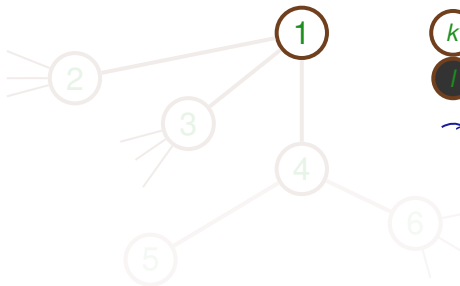
Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési
Feladatok



k – nyílt csúcsok

l – zárt csúcsok

$\rightarrow$ – $p()$ szülő



Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus
Általános algoritmus

Példa

Mélységi

Szélességi

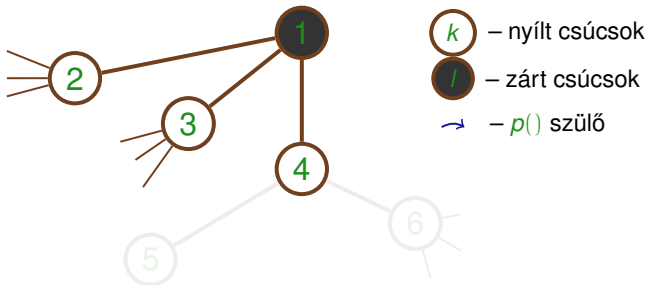
Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési
Feladatok





Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus
Általános algoritmus

Példa

Mélységi

Szélességi

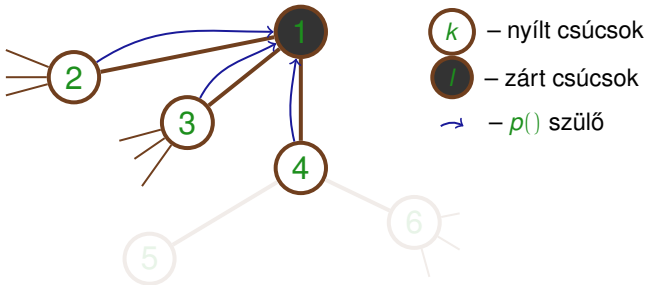
Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési
Feladatok





Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus
Általános algoritmus

Példa

Mélységi

Szélességi

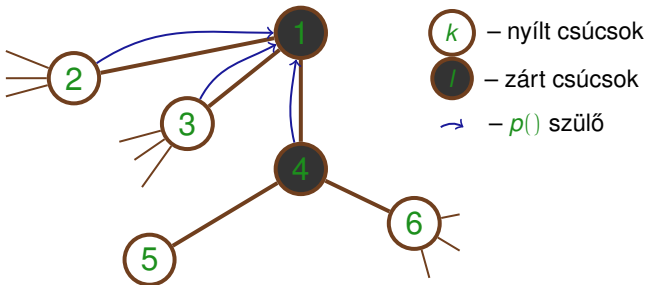
Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési
Feladatok



Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus
Általános algoritmus

Példa

Mélységi

Szélességi

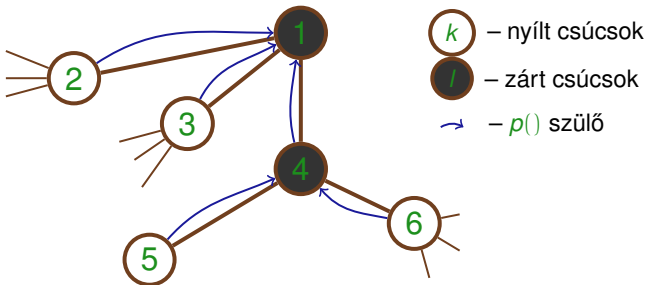
Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési
Feladatok



Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

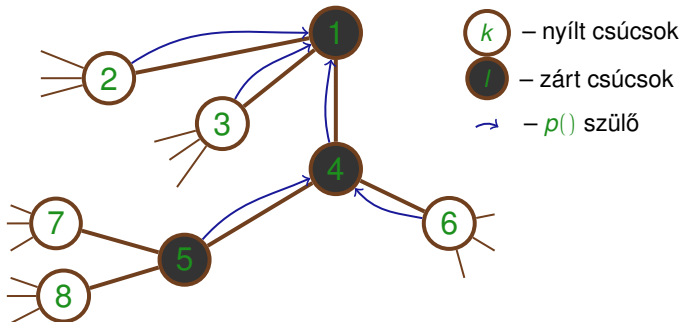
Előretekintő

A alg

A* és A_c

Gráfkeresési

Feladatok



Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

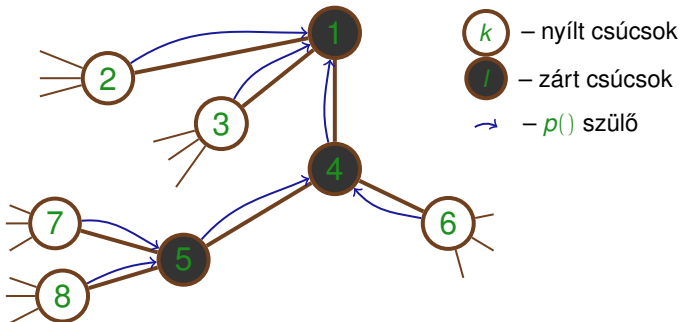
Előretekintő

A alg

A* és Ac

Gráfkeresési

Feladatok



Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus
Általános algoritmus

Példa

Mélységi

Szélességi

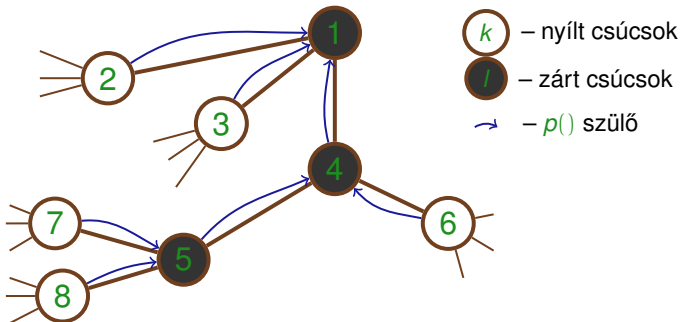
Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési
Feladatok



A **harmadik** iteráció végén tehát

- a zárt csúcsok halmaza $\{1, 4, 5\}$;
- a nyílt csúcsok halmaza $\{2, 3, 6, 7, 8\}$;
- a szülő-függvény $(z, p(z))$ párok:
 $\{(2, 1), (3, 1), (4, 1), (5, 4), (6, 4), (7, 5), (8, 5)\}$

Példa gráfkeresésre

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

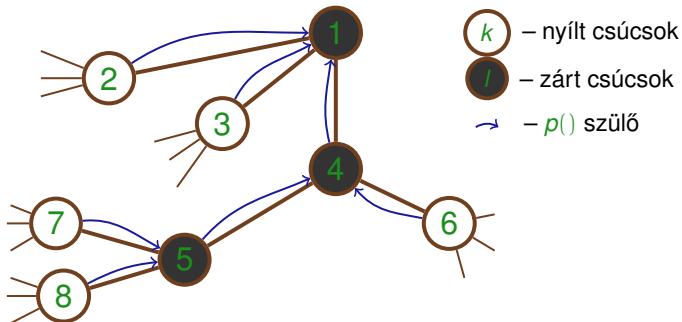
Előretekintő

A alg

A* és Ac

Gráfkeresési

Feladatok



A következő csúcs kiválasztása a nyílt halmazból **bármilyen** kritérium alapján történhet. A kritérium alapja az $f(\cdot)$ függvény.

A függvény megválasztásával különböző **keresési stratégiák**hoz jutunk.



Mélységi keresés

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus
Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési
Feladatok

- Mindig a legmélyebben fekvő nyílt csúcsot választjuk,
- Ha minden él költsége ugyanannyi (pl. $c(m, n) = 1$), akkor a kiértékelő függvény:

$$f(n) = -g(n) \quad \forall n \in NYILT,$$

- Szükséges (? – **mikor**) a mélységi korlát bevezetése,
- Az algoritmus nem mindig talál megoldást,
- Iteratív növelése a mélységi korlátnak – megoldást talál (?**optimális**?).



Szélességi keresés

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési

Feladatok

- A mélységi keresés ellentettje,
- Mindig a legmagasabban fekvő nyílt csúcsot választjuk,
- Ha minden él költsége ugyanannyi (pl. $c(m, n) = 1$), akkor a kiértékelő függvény:

$$f(n) = g(n) \quad \forall n \in NYILT$$

- Az algoritmus **mindig** talál megoldást – amennyiben ez létezik.



Egyenletes keresés

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési

Feladatok

- Súlyozott változata a szélességi keresésnek,
- A kiértékelő függvény (általános $c(m, n)$ esetén):

$$f(n) = g(n) \quad \forall n \in NYILT$$

- Az algoritmus **mindig** talál megoldást – amennyiben ez létezik,
- Dijkstra algoritmusa (1959).



Előrettekintő keresés

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előrettekintő

A alg

A* és Ac

Gráfkeresési

Feladatok

- Heurisztikus kereső algoritmus,
- A kiértékelő függvény **csak** a heurisztika:

$$f(n) = h(n) \quad \forall n \in NYILT$$

- $h(n)$ – heurisztikus függvény.
- pl. $f(n) = W(n)$ – a 8-as kirakójátékban;
- A keresőgráf kisebb, mint az egyenletes keresés esetében;
- A keresőgráf **nem optimális**;
- Erősen függ a választott keresőfüggvényről.



„A” algoritmus

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési

Feladatok

Futó: Mesterséges Intelligencia – pp. 90

„ötvözi az egyenletes keresés óvatosságát az előretekintő keresés célratörésével, egyesítve előnyös tulajdonságaikat”

- Kiértékelő függvény:

$$f(n) = g(n) + h(n) \quad \forall n \in NYILT$$

ahol $h(n) > 0$.

- $h(n)$ „becsüli” az n -ből a cél–csúcsba vivő optimális út költségét.



„A” algoritmus tulajdonságai

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési

Feladatok

Tulajdonságok

- $f^*(n) = g^*(n) + h^*(n)$ - a **start**ból a **cél**ba az n -en keresztül vivő optimális út költségének a becslése.
- Ha az **A** algoritmus **nem** terminál, akkor minden **NYILT** halmazba került csúcs véges sok lépés után kiterjesztésre kerül.
- Az **A** algoritmus mindig talál megoldást feltéve, hogy létezik megoldás.



„A*” algoritmus

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és A_c

Gráfkeresési

Feladatok

- A algoritmus kiértékelő függvénye, ahol
- A heurisztikus függvény bármely csúcsban alulról becsüli a a célba vezető optimális út költségét, azaz

$$h(n) < h^*(n) \quad \forall n \in G$$

- A fenti kritérium a heurisztika **megengedhetősége**.
Egy gráfkereső algoritmus megengedhető, ha megoldás létezése esetén megtalálja az optimális megoldást.

A* tulajdonságai

- Bármely kiterjesztésre választott csúcsra
 $f(n) \leq f^*(n)$.
- Mindig optimális megoldással terminál, feltéve ha az létezik.



„ A^c ” algoritmus

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A^* és A^c

Gráfkeresési

Feladatok

- **Korlátozás a heurisztikus függvényre:**

$h(n)$ kielégíti a monoton megszorítás (monotone restriction) feltételét, ha

$$h(n) - h(m) \leq c(n, m) \quad \forall (n, m) \in A$$

- Egy n csúcs **nem** kedvezőtlen, ha egy utód m csúcs nagyon kedvező.
- **A^c algoritmus**: az olyan A algoritmus, ahol $h(n)$ monoton megszorításos és $h(t) = 0$ minden terminális csúcsra.



„ A^* ” algoritmus tulajdonságai

Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A^* és A_c

Gráfkeresési

Feladatok

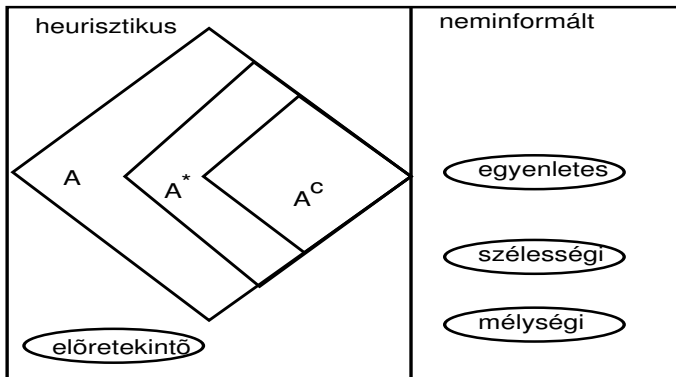
A^* tulajdonságai

- Ha teljesül a monoton megszorítás, akkor egy n csúcsba vezető optimális út mentén a $g + h$ növekvő.
- Bármely n kiterjesztésre választott csúcshoz az optimális út van megjelölve:

$$g(n) = g^*(n)$$

- Mindig optimális megoldással terminál, feltéve ha az létezik.

- a heurisztika **nagyon fontos** – egy algoritmus alkalmazhatósága a választott heurisztikán áll vagy bukik.





Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési
Feladatok

Particionálás:

Vizsgáljuk meg, hogy létezik-e az $\{1, \dots, N\}$, $N = 10$ halmaznak a következő tulajdonságokkal rendelkező k darab $k = 8$ részhalmaza, melyekre:

- 1 bármely két (2) elem legtöbb két (2) halmazban szerepel, illetve
- 2 bármely két (2) halmaz metszete legtöbb két (2) elemből áll.

<http://www.maths.qmw.ac.uk/~pjc/oldprob.html>

Feladat: Írjunk egy programot, mely megkeresi az első ilyen részhalmazrendszert. A program paraméterezhető kell, hogy legyen, a paraméterei az (N, k) páros.

Nagyobb teszt: létezik az $(N = 15, k = 12)$ esetre megoldás?

(15 pont)

Kötelező feladat



Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési
Feladatok

A háromjegyű számok halmazán értelmezzük a következő feladatot: jussunk el egy I számból egy adott G -hez a következő szabályok szerint:

- 1 Egy számjegyet eggyel növelünk vagy csökkentünk,
- 2 Egymásutáni lépésekben különböző a számjegyeket módosítunk,
- 3 A 9-es számjegyhez nem lehet hozzáadni; a 0-ból nem lehet kivonni,
- 4 Nem tehető olyan lépés, mely a tiltott halmazba tartozó számot állítana elő (a tiltott lista adott)

A feladat megoldásához használjuk az A^* algoritmust.

$$h(I) = |G_1 - I_1| + |G_2 - I_2| + |G_3 - I_3|$$

(5 pont)

Forrás: Szalay



Mesterséges
Intelligencia

4

Csató Lehel

Alapalgoritmus

Általános algoritmus

Példa

Mélységi

Szélességi

Egyenletes

Előretekintő

A alg

A* és Ac

Gráfkeresési
Feladatok

Legyen a következő „összeadás”:

$$\begin{array}{rcccc} & S & E & N & D \\ & M & O & R & E \\ \hline M & O & N & E & Y \end{array}$$

ahol:

- A betűk mindegyike számjegy.
- „értelmes” számok: *S* és *M* nem lehet 0.

Feladat: Írjunk egy programot, mely a keresést végrehajtja.

(3 pont)

Forrás: Szalay