A mesterséges intelligencia alapjai

informált keresések

Áttekintés

- mohó legjobbat-először keresés
- A* keresés
- heurisztikák

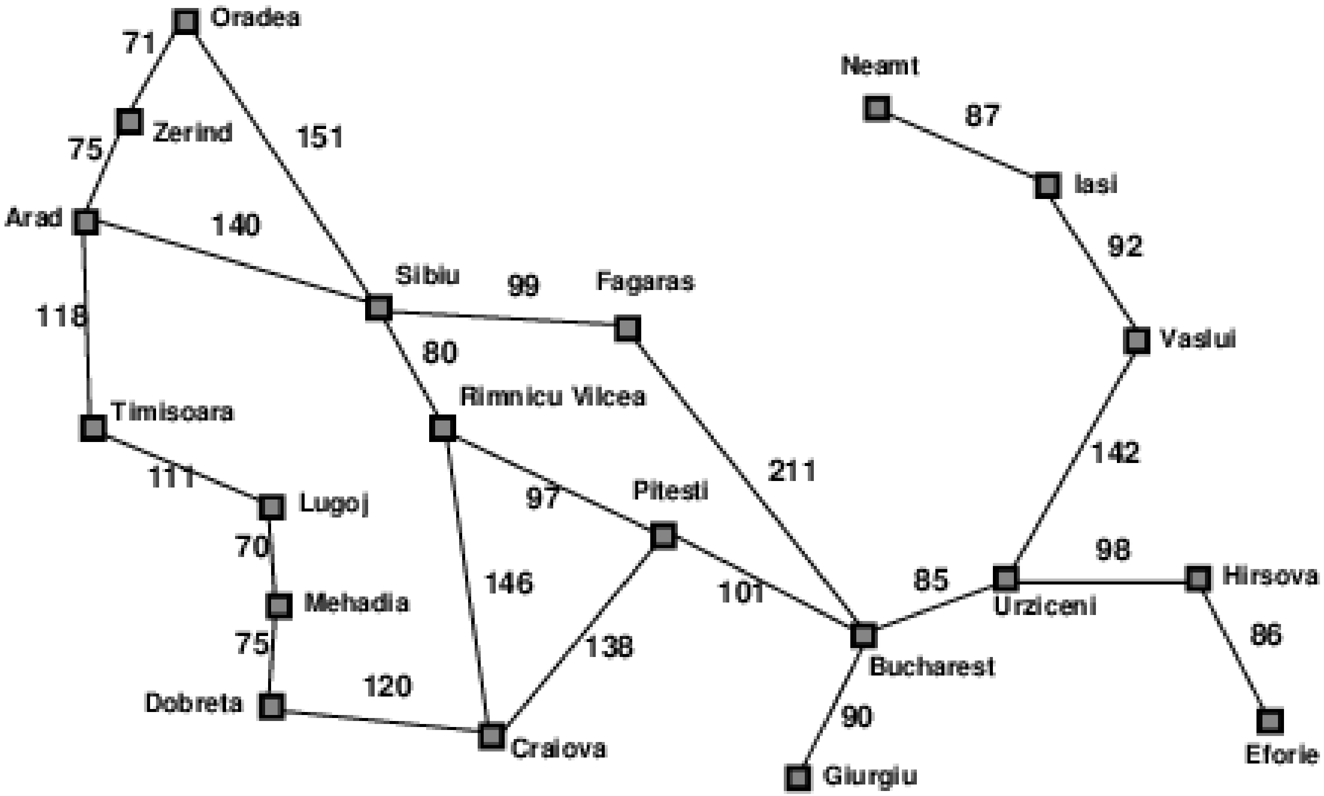
Fakereséső algoritmus (ismétlés)

```
function Tree-Search(problem, strategy): megoldás vagy “sikertelen”
    kereső fa inicializálása a probléma kezdőállapotával
    loop do
        if nincs kiterjeszthető csúcs
            then return “sikertelen”
        válassz a stratégia alapján egy levél csúcsot
        if a csúcs célállapotot tartalmaz
            then return kapcsolódó megoldást
        else terjeszd ki a csúcsot, és a gyerekcsúcsokat add a keresőfához
    end
```

Heurisztikus keresés

- alapötlet
  - használjunk egy **kiértékelő függvényt**, mely a csúcsokon van értelmezve
  - a csúcs *kívánatosságát* becsüli meg
  - a leginkább kívánatos csúcsot terjesszük ki
- implementáció
  - a perem egy sor, mely a kívánatosság szerint rendezett
- speciális esetek
  - mohó legjobbat-először keresés
  - A* keresés

Románia térképe, és a Bukaresttől mért távolságok



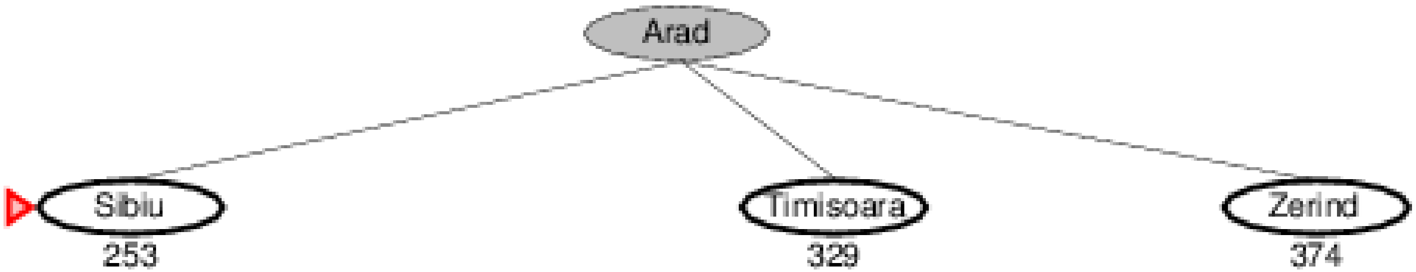
Straight–line distance to Bucharest	
Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Mohó legjobbat-először keresés

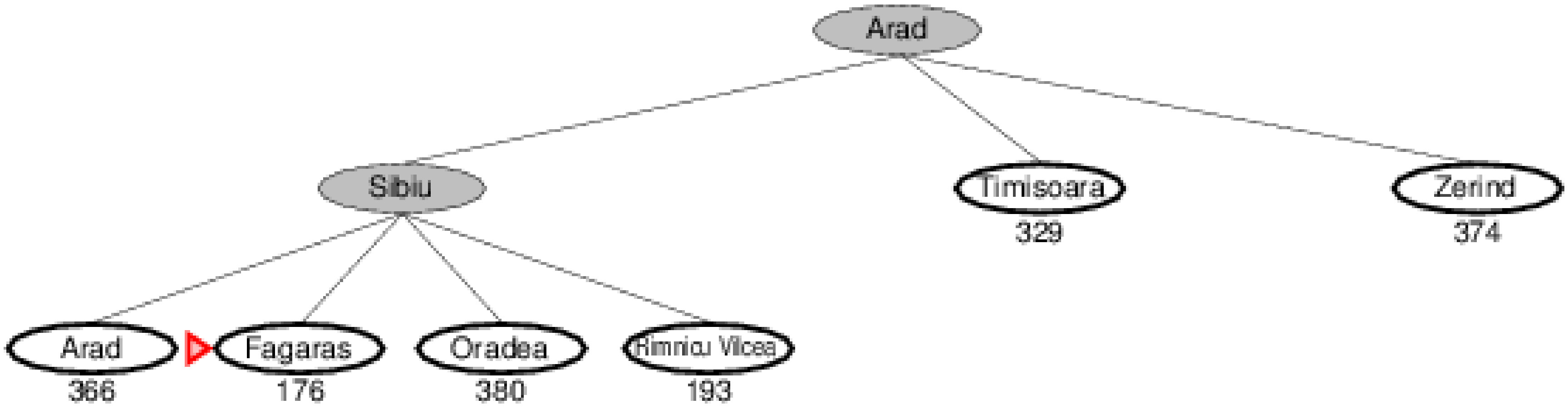
- kiértékelő függvény: *h(n)*
  - heurisztika
  - megbecsüli *n* távolságát a legközelebbi céltól

Példa mohó legjobbat először keresésre

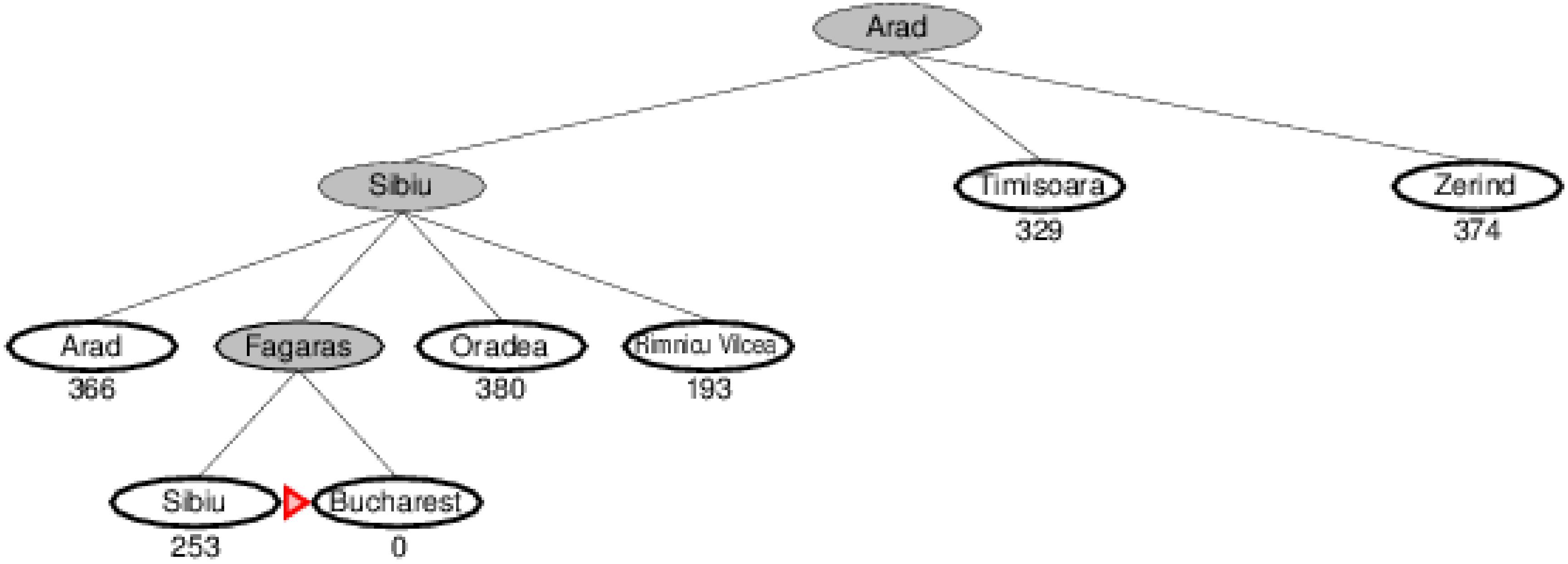
Példa mohó legjobbat először keresésre



Példa mohó legjobbat először keresésre



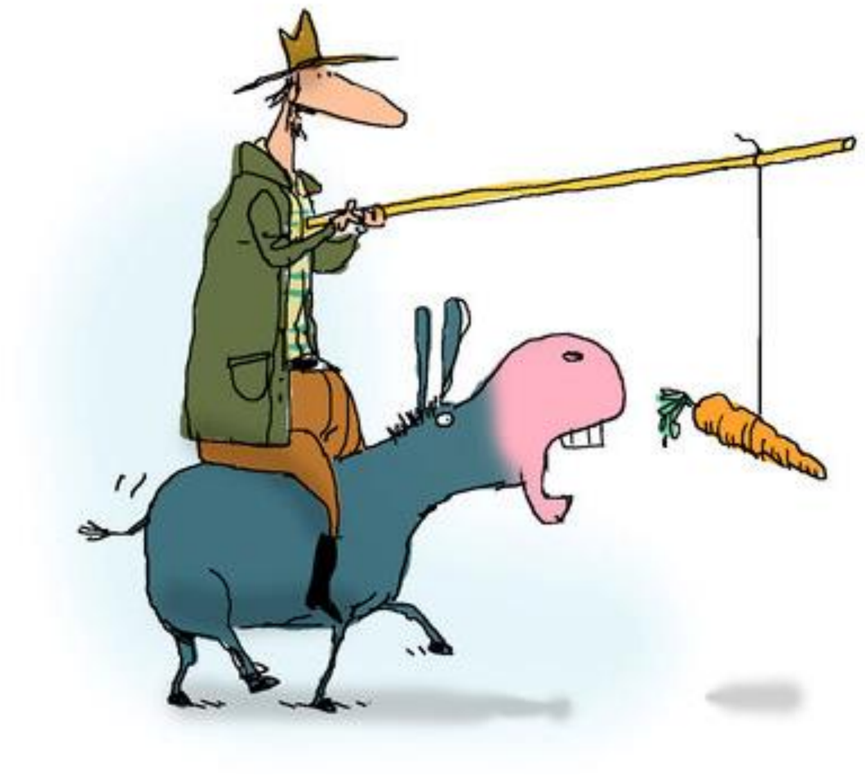
Példa mohó legjobbat először keresésre



A mohó legjobbat-először keresés tulajdonságai

- teljesség
  - nem, beragadhat
    - Nagyvárad céllal: Jászvásár (Iasi) → Karácsonkő (Neamt) → Jászvásár → Karácsonkő → …
  - véges állapottér esetén, ismételt állapotok vizsgálatával teljessé tehető
- időbonyolultság
  - $O(b^m)$, de egy jó heurisztika drámaian felgyorsíthatja
- tárbonyolultság
  - $O(b^m)$, minden csúcsot a memóriában tart
- optimalitás
  - nem

A mohó legjobbat-először keresés és az optimalitás

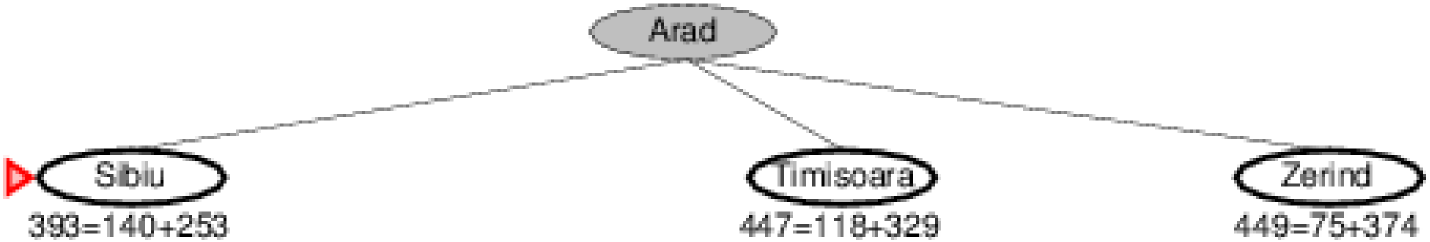


A* keresés

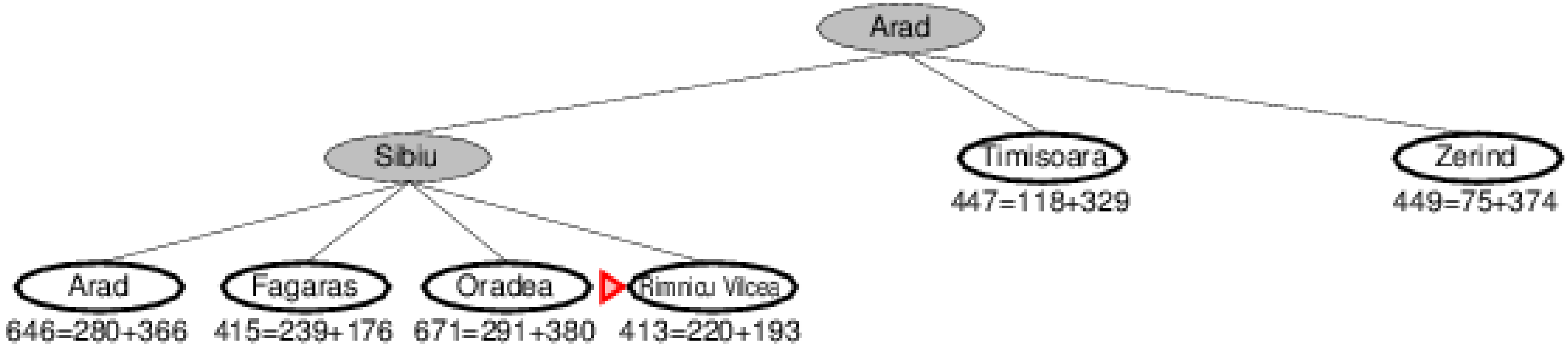
- ötlet
  - ne terjesszük ki azokat az utakat, melyek már eleve drágák
- kiértékelő függvény $f(n) = g(n) + h(n)$
  - $g(n)$ – útköltség n-ig
  - $h(n)$ – a célig tartó út becsült költsége n-től
  - $f(n)$ – az n-en keresztül a célba vezető út becsült teljes költsége
- A* keresés – **elfogadható** heurisztikát használva
  - $h(n) \leq h^*(n)$, ahol $h^*(n)$ a *valós* költség a célig
  - $h(n) \geq 0$, így $h(c)=0$ minden c cél esetén
  - a légvonalban mért távolság nem becsüli felül az úton mért távolságot
- Tétel: Az A* fakeresés optimális

Példa A* keresésre

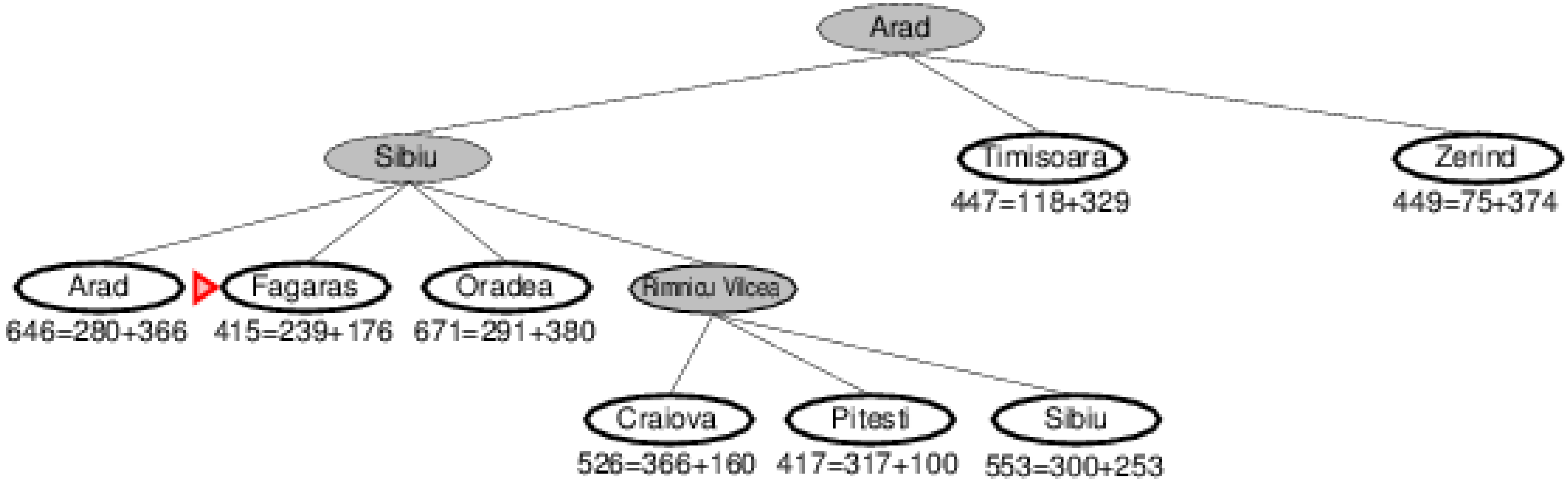
Példa A* keresésre



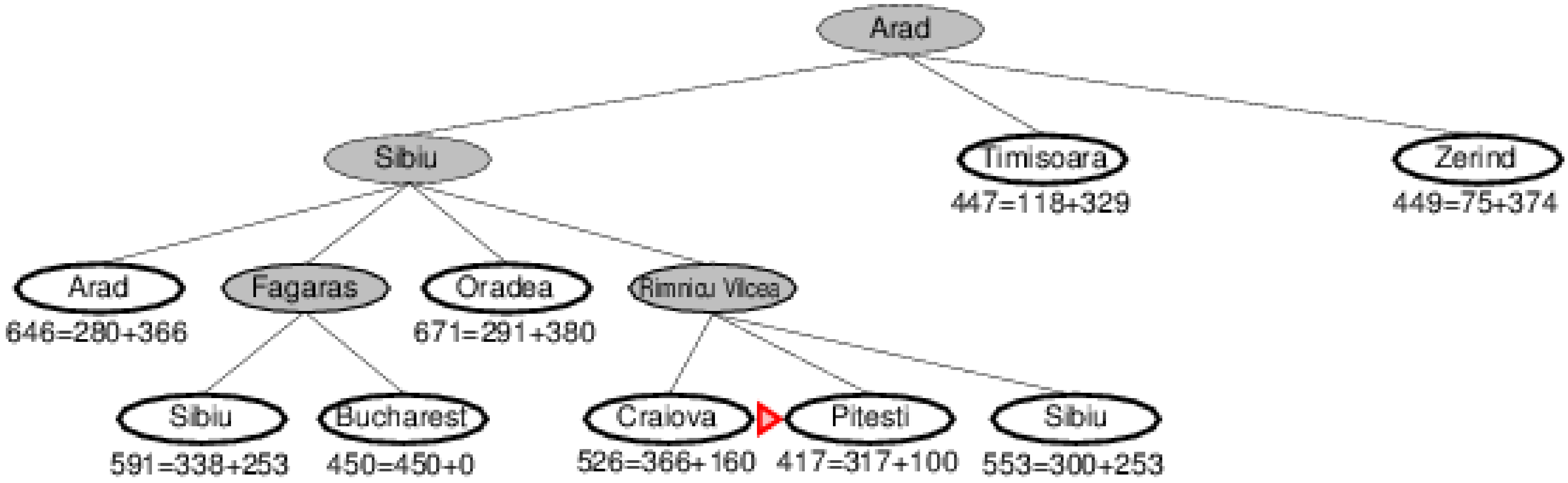
Példa A* keresésre



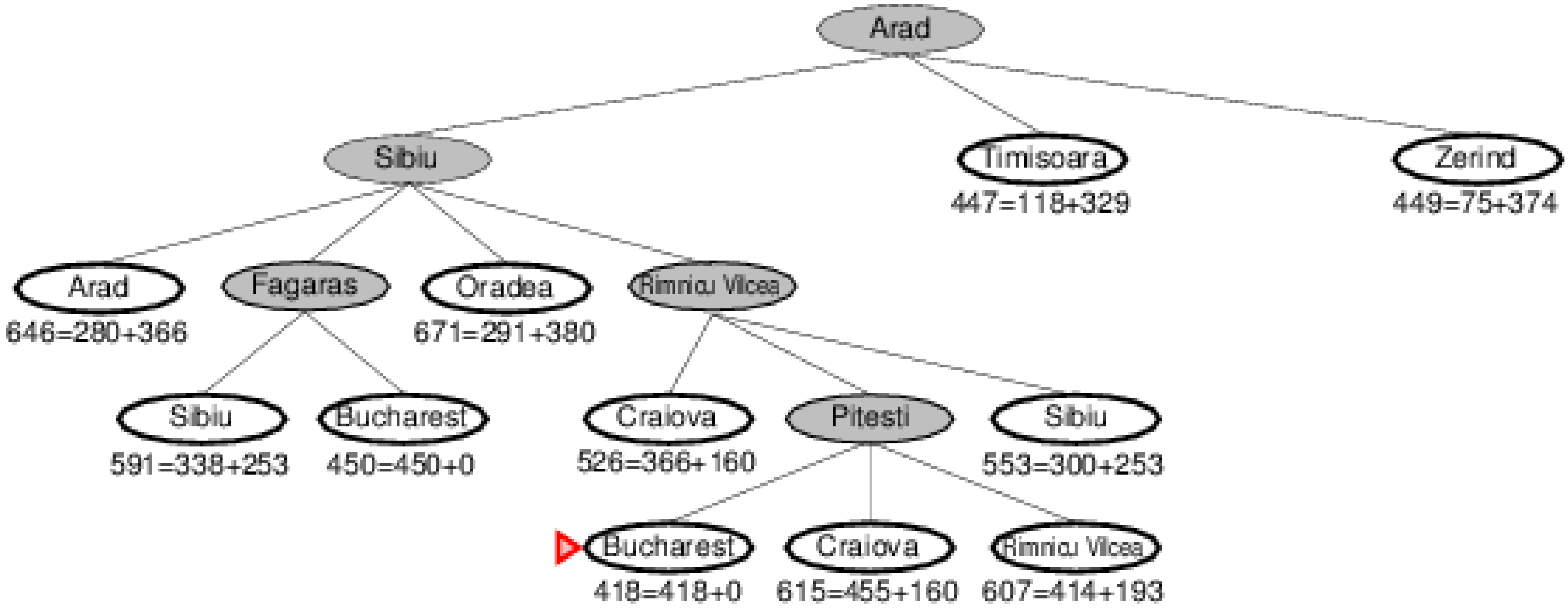
Példa A* keresésre



Példa A* keresésre



Példa A* keresésre

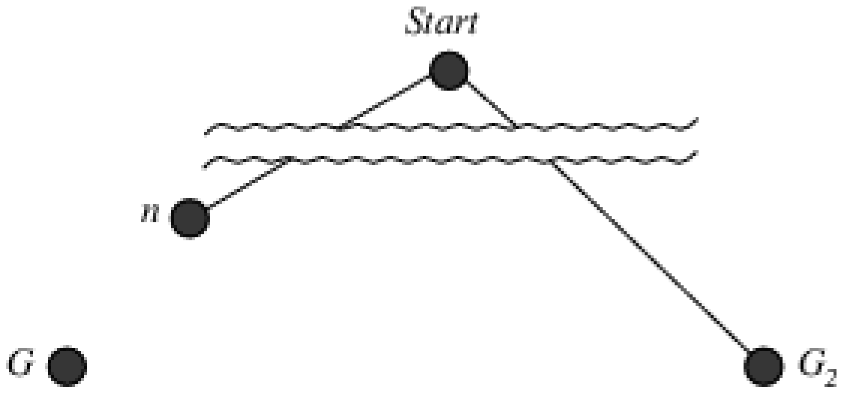


A* optimalitása (standard bizonyítás)

Tegyük fel, hogy a peremen egy G_2 szub-optimális célcsomópont jelenik meg, és az optimális megoldás útköltsége C. mivel G_2 szuboptimális $f(G_2)=g(G_2)+h(G_2) > C$.

Legyen n egy csúcs a peremben, mely az optimális megoldás útvonalán fekszik, Ha h(n) nem becsüli túl a valós költséget, akkor $f(n)=g(n)+h(n)\leq g(n)+h^*(n)=C$.

Így G_2 nem kerül kifejtésre, így A* optimális megoldást ad.

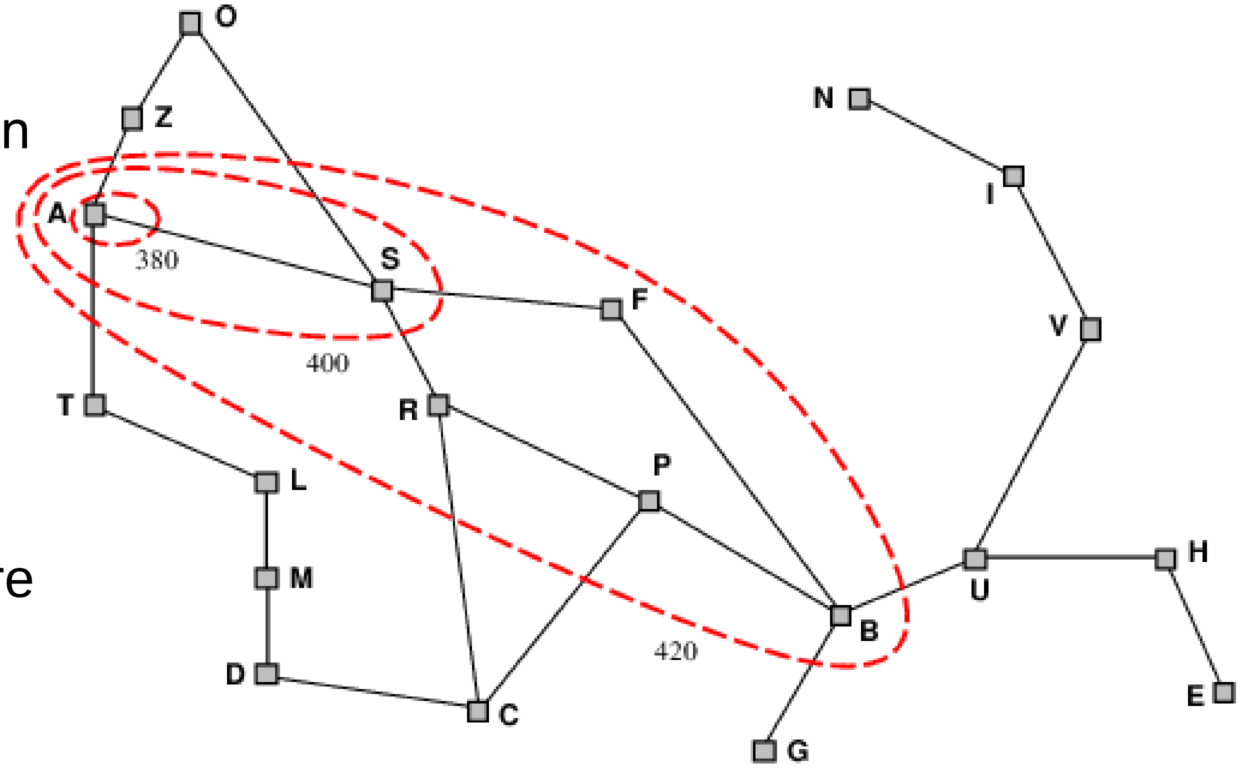


A* optimalitása

Lemma: A* a csúcsokat növekvő f értékük alapján terjeszti ki.

Fokozatosan építi az f-kontúrokat.

Az i. kontúr tartalmazza az összes csúcsot melyre $f = f_i$, ahol $f_i < f_{i+1}$



A* fakeresés tulajdonságai

- teljesség
  - igen, ha nincs végtelen sok csúcs, melyre $f \leq f(G)$
- időbonyolultság
  - exponenciális (h relatív hibája × megoldás hossza)
- tárbonyolultság
  - minden csúcsot a memóriában tart
- optimalitás
  - igen, nem bontja ki f_{i+1}-et, amíg f_i-vel nem végez
  - A* kibont minden csúcsot, melyre $f(n) < C$
  - A* kibont néhány csúcsot, melyre $f(n) = C$
  - A* nem bont ki csúcsot, melyre $f(n) > C$

Konzisztens heurisztika

Egy heurisztika **konzisztens** (monoton), ha

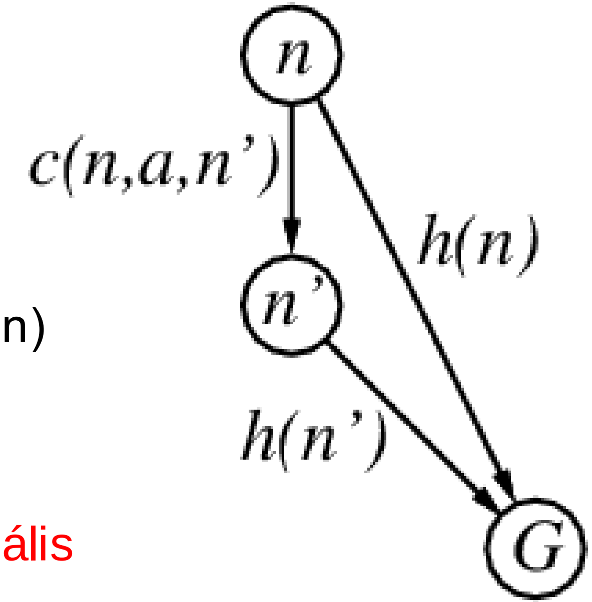
$$h(n) \leq c(n,a,n') + h(n')$$

Ha h konzisztens, akkor

$$f(n') = g(n')+h(n') = g(n)+c(n,a,n')+h(n') \geq g(n)+h(n) = f(n)$$

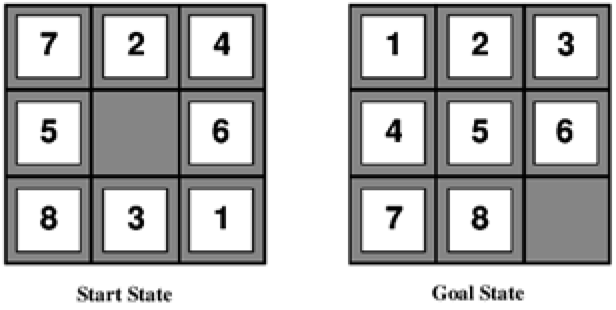
Így f(n) monoton nemcsökkenő minden út mentén

Konzisztens heurisztika esetén a A* gráfkeresés optimális



Elfogadható heurisztikák – nyolcas játék

- $h_1(n)$ – rossz helyen álló lapok száma
- $h_2(n)$ – teljes Manhattan távolság (vízszintes és függőleges távolságok összege)
- $h_1(S) = 6$
- $h_2(S) = 4+0+3+3+1+0+2+1 = 14$



A megoldás 26 lépésből áll

Heurisztikák összehasonlítása

ha $h_2(n) \geq h_1(n)$ minden n csúcsra (és mindkettő elfogadható heurisztika), akkor h_2 **dominálja** h_1-et, és sokkal hasznosabb keresésre

d=14	iteratívan mélyülő	3 473 941
	A* – h_1	539
	A* – h_2	113
d=24	iteratívan mélyülő	$\simeq$ 54 000 000 000
	A* – h_1	39 135
	A* – h_2	1 641

Dominancia felhasználása

Ha adott két elfogadható heurisztika h_a és h_b, akkor

$$h(n) = \max(h_a(n), h_b(n))$$

szintén elfogadható heurisztika, és dominálja h_a-t és h_b-t is.

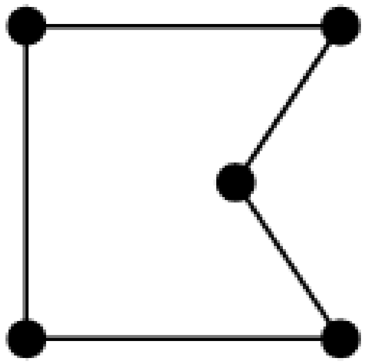
Relaxált problémák

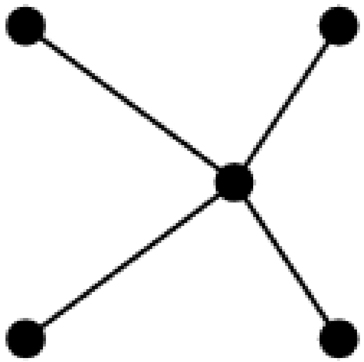
- azt a problémát, melyben az operátorokra kevesebb megkötést teszünk mint az eredeti problémában, **relaxált problémának** nevezzük
- elfogadható heurisztika konstruálható a relaxált probléma pontos megoldási költsége alapján
- ha a nyolcas játékban egy lapot *bárhova* rakhatunk, $h_1(n)$ adja meg a legrövidebb megoldást
- ha a nyolcas játékban egy lapot bármely szomszédos mezőre áttolhatunk, akkor $h_2(n)$ adja meg a legrövidebb megoldást.
- fontos: a relaxált probléma optimális megoldásának költsége nem nagyobb, mint az eredeti probléma optimális megoldásának költsége

Relaxált problémák

Az **utazó ügynök probléma** (TSP) egy közismert feladat: adjuk meg a legrövidebb utat, mellyel minden várost pontosan egyszer látogatunk meg (minimális Hamilton-kör)

A minimális feszítőfa $O(n^2)$ idő alatt határozható meg, s ez egy alsó korlátot ad meg a problémára (Held, Karp 1970)





Összefoglalás

- a heurisztika függvények a legrövidebb utak hosszát becsülik
- egy jó heurisztika jelentősen lerövidíti a keresést
- a mohó legjobbat-először keresés a legkisebb heurisztikájú csúcsot terjeszti ki
    - nem teljes és nem feltétlenül optimális
- A* keresés a legkisebb g+h-t terjeszti ki
    - teljes és optimális
    - optimálisan hatékony
        - egyetlen más algoritmus sem fejt ki garantáltan kevesebb csomópontot
- elérhető heurisztikák nyerhetők a relaxált feladatok pontos megoldásaiból