

2. Seminarska naloga

UMETNA INTELIGENCA

1. Neinformirano preiskovanje

Sistematično preiskovanje prostora. Preiščemo vse možne kombinacije stanj.

- DFS

Iskanje v globino (Depth-first search). Uporaba sklada za pregled elementov. Da algoritem ne išče v globino v nedogled sem uporabil mejo za maksimalno globino iskanja (maxDepth). Algoritem najde rešitev **hitro**, z **malo število elementov v pomnilnik**, ampak najdena pot **NI OPTIMALNA**.

- BFS

Iskanje v širino (Breadth-first search). Uporaba vrste za pregled elementov. Algoritem najde rešitev **počasneje** kot DFS, ker mora pregledat veliko več stanj (vsa stanja do določene globine). V **pomnilniku je lahko naenkrat zelo veliko elementov** (število elementov na globini). Najdena pot je **OPTIMALNA**.

- IDDFS

Iterativno poglobljanje Iskanja v globino (ID Depth-first search). Uporaba rekurzije za pregled elementov. Algoritem pregleduje stanja v globino, pri tem se **globina iterativno povečuje**. MaxDepth se poveča, ko preiščemo vsa stanja na trenutni globini. Ker iščemo v globino, je **max število elementov v pomnilniku (globina)**. V primerjavi z BFS imamo veliko manj elementov v pomnilniku, zato lahko s tem algoritmom preiščemo **večji prostor**. Cena tega je, da moramo **ista stanja večkrat pregledati** (število vseh pregledanih stanj). Najdena pot je **OPTIMALNA**.

2. Hevristično preiskovanje

Da prostor preiskanih stanj zmanjšamo, uporabimo hevristične ocene. Ker smo prostor stanj, ki jih moramo preiskati zmanjšali, bodo vsi algoritmi z uporabo hevrstike veliko hitreje našli rešitev, kot algoritmi s sistematičnim preiskovanjem prostora.

- GBFS

Požrešno iskanje (Greedy Best-first search). Uporaba prioritete vrste za pregled elementov. Algoritem najde rešitev zelo **hitro**, z **malo številom elementov v pomnilnik**, ampak najdena pot **NI OPTIMALNA**.

- A*

Uporaba prioritete vrste za pregled elementov. Število vseh pregledanih stanj je malo, a je zelo veliko elementov hkrati v pomnilniku. Najdena pot je **OPTIMALNA**.

- IDA*

Iterativno poglobljanje A*. Uporaba prioritete vrste in rekurzije za pregled elementov. Algoritem deluje kot A*, dodatno se beleži meja »bound«, ki se postopoma povečuje za pregled stanj z večjo hevristično oceno. Ker iščemo, do dane meje »bound«, se **max število elementov v pomnilniku (globina) drastično zmanjša** v primerjavi z A*. Zato lahko s tem algoritmom preiščemo **večji prostor**. Cena tega je, da moramo **ista stanja večkrat pregledati** (Število vseh pregledanih stanj). Najdena pot je **OPTIMALNA**.

3. Hevristična ocean

- **Ocena 1:** Hevristika stanja se določi po pregledu pravih položajev škatel. Če imamo škatlo na napačnem mestu, bomo potrebovali vsaj se 1 premik, da bo a pravem mestu.
- **Ocena 2:** Hevristika je mal obol napredna. Deluje enako kot Ocena 1 z dodatkom, da za pravilno pozicionirano škatlo pregleda škatle pod njo. V primeru, da je skala pod njo na napačnem mestu, moramo trenutno škatlo premakniti s pozicije, na pozicijo dodati pravilno škatlo in trenutno škatlo dodati nazaj na pravo pozicijo. Pomeni da bomo potrebovali se vsaj 2 dodatna premika.

Primer 4 (DFS vs BFS vs IDDFS)

----- DFS -----

Dolzina poti: 30

Stevilo vseh pregledanih stanj: 38547

Max ele v pomnilniku (stack): 31

Porabljen cas: 40 ms

----- BFS -----

Dolzina poti: 12

Stevilo vseh pregledanih stanj: 32399

Max ele v pomnilniku (queue): 10691

Porabljen cas: 71 ms

----- IDDFS -----

Dolzina poti: 12

Stevilo vseh pregledanih stanj: 160663

Max ele v pomnilniku (rekurzija): 12

Porabljen cas: 87 ms

Primer 4 (GBFS vs A* vs IDA*)

----- GBFS -----

Dolzina poti: 13

Stevilo vseh pregledanih stanj: 19

Max ele v pomnilniku: 107

Porabljen cas: 1 ms

----- A* -----

Dolzina poti: 12

Stevilo vseh pregledanih stanj: 5587

Max ele v pomnilniku: 8305

Porabljen cas: 18 ms

----- ID A* -----

Dolzina poti: 12

Stevilo vseh pregledanih stanj: 18327

Max ele v pomnilniku: 25

Porabljen cas: 15 ms

Primer 5 (DFS vs BFS vs IDDFS)

----- DFS -----

Dolzina poti: 29

Stevilo vseh pregledanih stanj: 72

Max ele v pomnilniku (stack): 31

Porabljen cas: 5 ms

----- BFS -----

Dolzina poti: 8

Stevilo vseh pregledanih stanj: 99361

Max ele v pomnilniku (queue): 53505

Porabljen cas: 330 ms

----- IDDFS -----

Dolzina poti: 8

Stevilo vseh pregledanih stanj: 184498

Max ele v pomnilniku (rekurzija): 8

Porabljen cas: 162 ms

Primer 5 (GBFS vs A* vs IDA*)

----- GBFS -----

Dolzina poti: 10

Stevilo vseh pregledanih stanj: 18

Max ele v pomnilniku: 163

Porabljen cas: 1 ms

----- A* -----

Dolzina poti: 8

Stevilo vseh pregledanih stanj: 216

Max ele v pomnilniku: 1259

Porabljen cas: 3 ms

----- ID A* -----

Dolzina poti: 8

Stevilo vseh pregledanih stanj: 1051

Max ele v pomnilniku: 17

Porabljen cas: 2 ms