

Tema 1 - IA: Sokoban

1. Implementarea algoritmilor si testare

Am implementat ambii algoritmi ceruti: IDA* si Beam Search. Fiecare a fost testat pe minim doua harti diferite, de complexitate crescatoare (easy_map1.yaml, medium_map2.yaml, hard_map2.yaml, etc.).

2. Jurnalizare euristici

Initial, pentru IDA*, am folosit o euristica bazata pe distanta Manhattan intre fiecare cutie si tinta cea mai apropiata. Pe unele harti (ex: medium_map2), jucatorul ramanea blocat in bucle de explorare. Pentru a remedia acest lucru, am introdus o penalizare pentru cutiile blocate in colturi. Comportamentul s-a imbunatatit considerabil – IDA* reusea sa evite starile fara iesire. Pentru Beam Search, am folosit aceeasi abordare.

3. Optimizari specifice implementarii

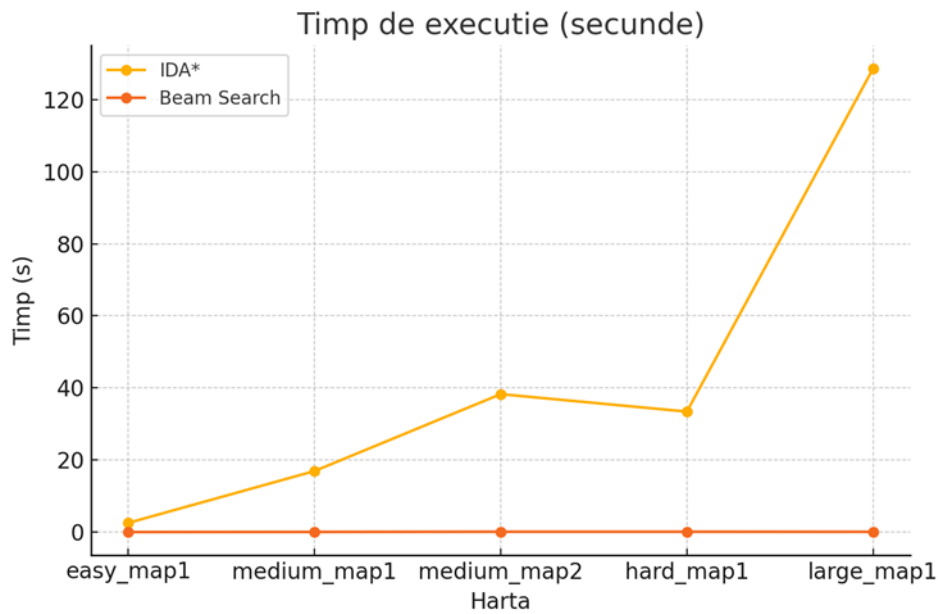
- Am eliminat miscarile de tip 'pull', care nu erau dorite in configuratia temei.
- In IDA*, am implementat o varianta imbunatatita a DFS bazata pe pseudocodul clasic, cu backtracking si memorare.
- In Beam Search, am introdus sortare explicita si control asupra dimensiunii frontierei pentru stabilitate si performanta.
- Am contorizat numarul de stari explorate si timpul de executie pentru analiza comparativa.

4. Comparatie algoritmi

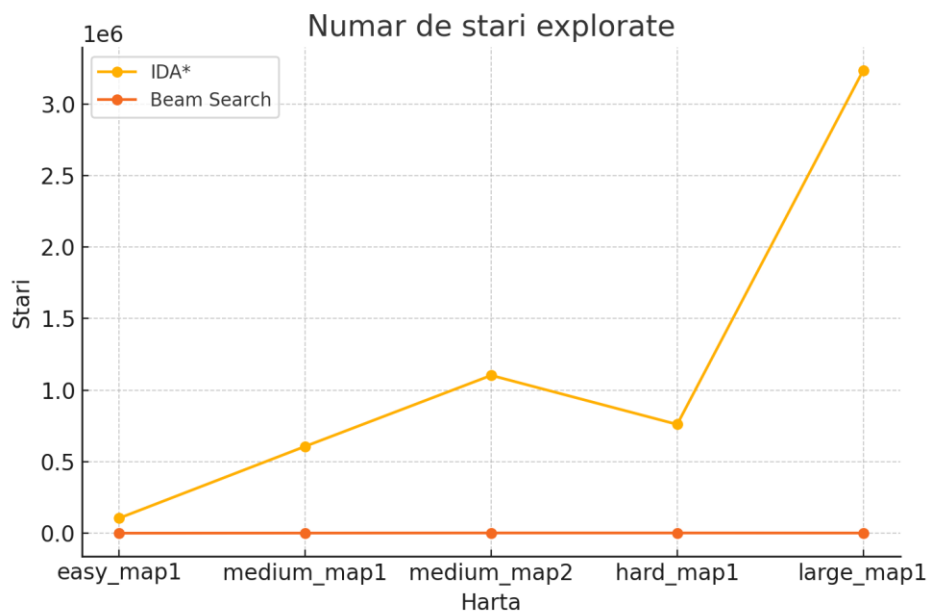
Comparatia a fost realizata pe 5 harti diferite si a fost structurata pe trei criterii:

- Timp de executie
- Numar de stari explorate
- Calitatea solutiei

Timpe de executie



Numar de stari explorate



5. Concluzii

Beam Search este semnificativ mai rapid, dar nu garanteaza mereu solutii optime. IDA* ofera rezultate corecte, dar necesita mult mai mult timp si explorare.

In practica, Beam Search este preferabil pentru harti mari sau complexe ca hard_map2 sau super_hard_map , iar IDA* este bun pentru testare precisa pe harti mici ca easy si medium.