Alexe Alexandra Florentina
ACS, grupa: 333 CC

Document de analiză - TEMA 1 - IA

Reprezentarea stărilor și a restricțiilor

Pentru algoritmul Hill Climbing, mi-am creat o clasa State care primea o calea către fișierul de intrare, un schedule și 3 dicționare necesare în prelucrare. Schedule reprezintă de fapt o variantă a orarului la un moment de timp, acesta fiind reprezetat mai exact printr-un dictionar de dictionare: cheia primului dictionar este ziua, iar valoarea este un alt dictionar. Cel de-al doilea dictionar cuprinde intervalul si o lista cu salile si ocuparea fiecareia de un profesor si o materie. Acest prim orar de la care incepe algoritmul este construit astfel incat sa respecte toate constrangerile hard in mod automat, iar ulterior prin varianta standard a algoritmului Hill Climbing, se imbunatateste de asemenea numarul constrangerilor hard. Reprezentarea restrictiilor este reflectata in cele 3 dictionare ajutatoare (informatii despre profesori, materii si sali) deoarece folosirea lor este esentiala in functia get_next_states prin care sunt gasite toate starile viitoare posibile.

In ceea ce priveste algoritmul Monte Carlo Tree Search, reprezentarea starilor si a restrictiilor este identica cu precizarea ca am modificat numele functiilor pentru a fi sugestive acestui algoritm. De asemenea, am incercat o abordare diferita, renuntand la utilizarea unei clase.

Explicarea detaliata a implementarii se regaseste in fisiere de cod. Am grupat functiile folosite in cele 2 abordari in 2 fisiere separate mcts.py si hc.py, iar in orar.py se gaseste functia principala. Asa cum este indicat in cerinta, codul se ruleaza utilizand fisierul orar.py, specificand fisierul de input si tipul de algoritm.

Rezolvarea folosind Hill Climbing - varianta standard

Consider ca este important de mentionat faptul ca am utilizat in generarea starilor, functii precum random_choice deoarece daca nu as fi folosit, algoritmul ar fi luat de fiecare data profesorii in aceeasi ordine. Dar inainte de acest pas, am

considerat o strategie buna alegerea profesorilor dupa anumite criterii, in limita posibilitatii. Cu alte cuvinte, cand a existat posibilitatea, am prioritizat in alegere profesorii care preferea anumite zile si intervale, si am prioritizat de asemenea profesorii care predau un numar mai mic de materii decat altii, acestia fiind mai putin versatili.

Am urmat varianta standard a algoritmului prin care sunt evitate punctele de minim local, aceasta varianta potrivindu-se cel mai bine abordarii mele.

Un punct forte al acestei abordari este faptul ca eu nu caut acoperirea unei materii intr-o sala care nu permite acest lucru. In schimb, este ceva ce as fi putut face mai bine pentru a avea costuri mai mici: as fi putut sa opresc parcurgerea in momentul in care toate materiile au fost acoperite, lucru care la mine nu se intampla deoarece am nevoie ulterior de o evidenta a salilor goale.

Rezolvare folosind Monte Carlo Tree Search

Asemenator cu abordarea de rezolvare folosind Hill Climbing, prioritizez in alegere profesorii in functie de numarul de materii pe care le pot preda si in functie de preferinte. Acest lucru reprezinta un criteriu in alegerea lor initiala, iar in cazul in care criteriul nu poate fi indeplinit se foloseste o alegere random, urmand ca numarul de conflicte se fie diminuat prin actiunile posibile. De aceasta data functia apply_move returneaza o lista cu actiunile valide asupra orarului. Aceasta functie asigura mentinerea unui cost 0 in ceea ce priveste hard constraints. Cele 3 posibile mutari luate in considerere (prezente de altfel si in implementarea HC) sunt: cautarea unei sali libere pentru a se sustine ora, schimbarea profesorului cu unul care nu si-a atins numarul maxim de ore predate sau schimbarea a doi profesori care pot preda fiecare in sala celuilalt, cu conditia ca salilele sa aiba aceeasi dimensiune.

Am folosit functiile de baza, specifice acestui algoritm, adaugand modificari specifice problemei in cauza, precum faptul ca am inlocuit functia care verifica daca o stare este finala, cu alta functie care se foloseste de cost-ul starii. Alta modificare a fost facuta la partea cu evaluarea recompenselor, deoarece am tinut cont de numarul de conflicte pentru a indica daca acea stare merita sa fie explorata si la viitoarele iteratii.

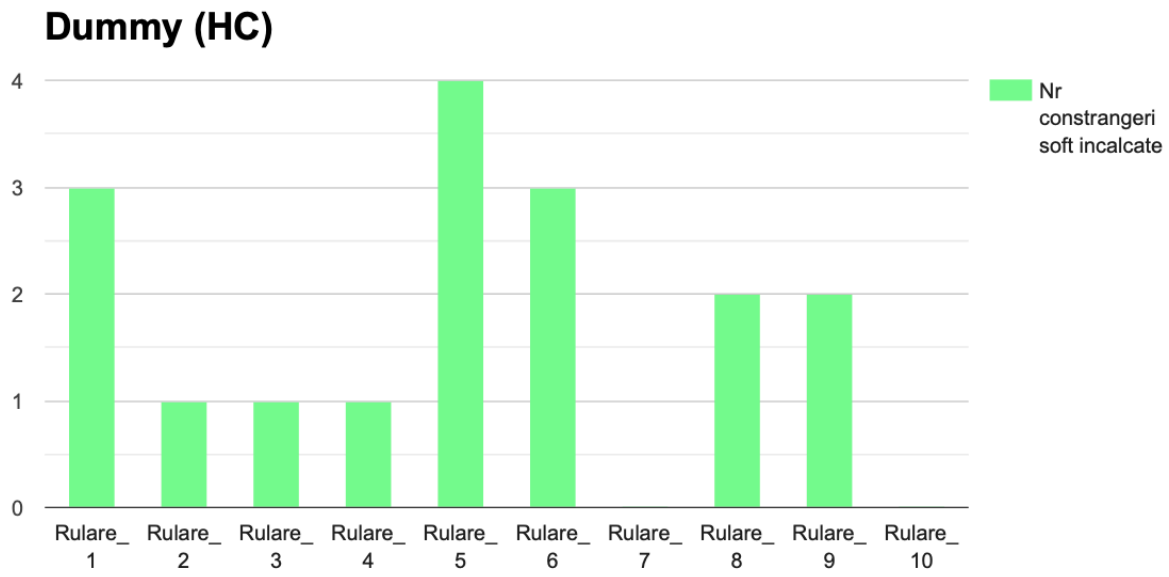
Comparația algoritmilor

Am comparat cei doi algoritmi în ceea ce privește numărul de constrângeri încălcate, numărul de stări expandate și timpul de execuție. Am considerat cele 5 orare date (excluzând-ul pe cel pentru bonus) și am rulat de 10 ori pentru fiecare. Numărul de

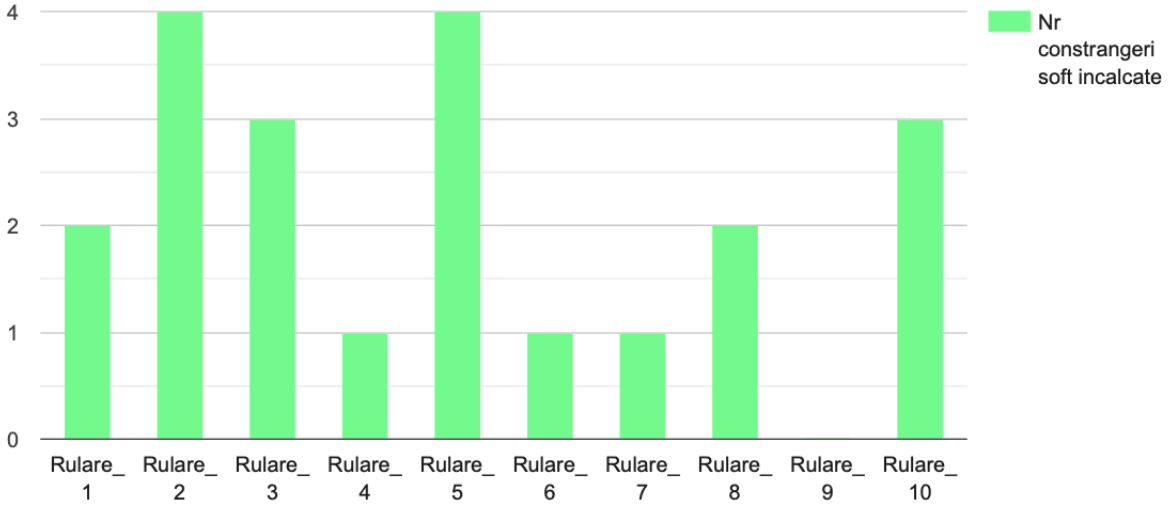
iterații l-am considerat 1000. În lipsa coloanei colorate e unei rulări, se consideră valoarea 0.

Trebuie sa afirm ca abordare cu HC are rezultate mai bune decat cea cu MCTS deoarece consider ca nu am gasit o euristica suficient de buna pentru MCTS. De asemenea, as mai fi putut face modificarii asupra algoritmului de baza mcts pentru a se potrivi problemei orarelor. Consider ca numarul de stari expandate este unul potrivit pentru abordarile folosite, iar timpul de executie este unul bun dat fiind faptul ca am specificat un numar maxim de iteratii.

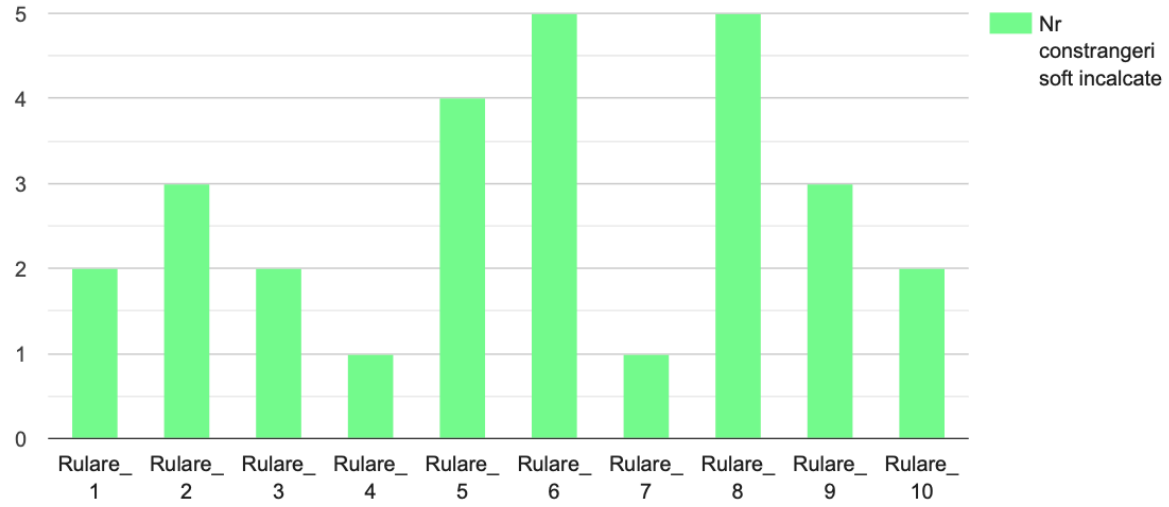
1. **Numărul de constrângeri încălcate**



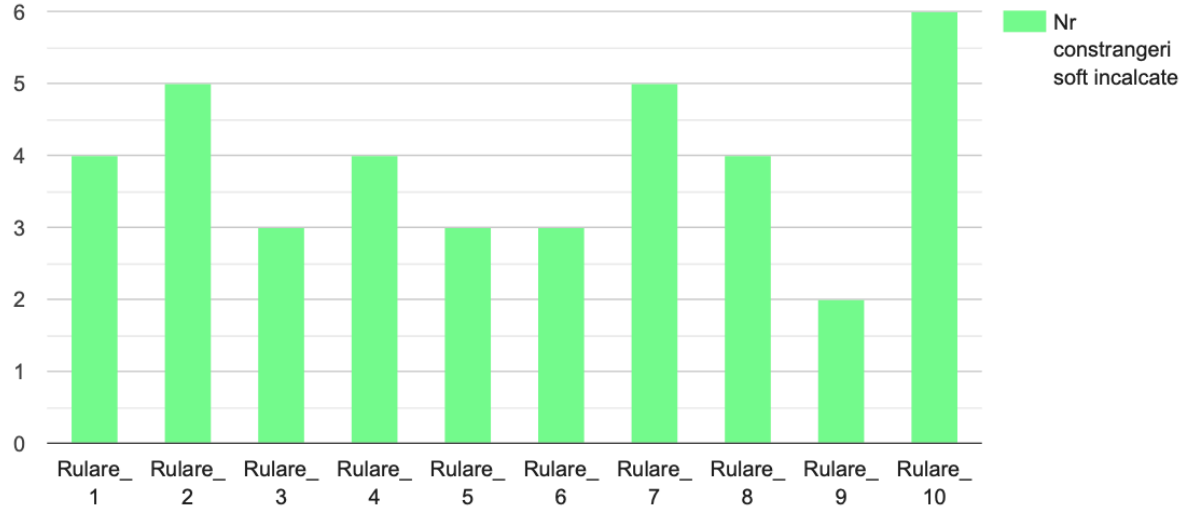
Dummy (MCTS)



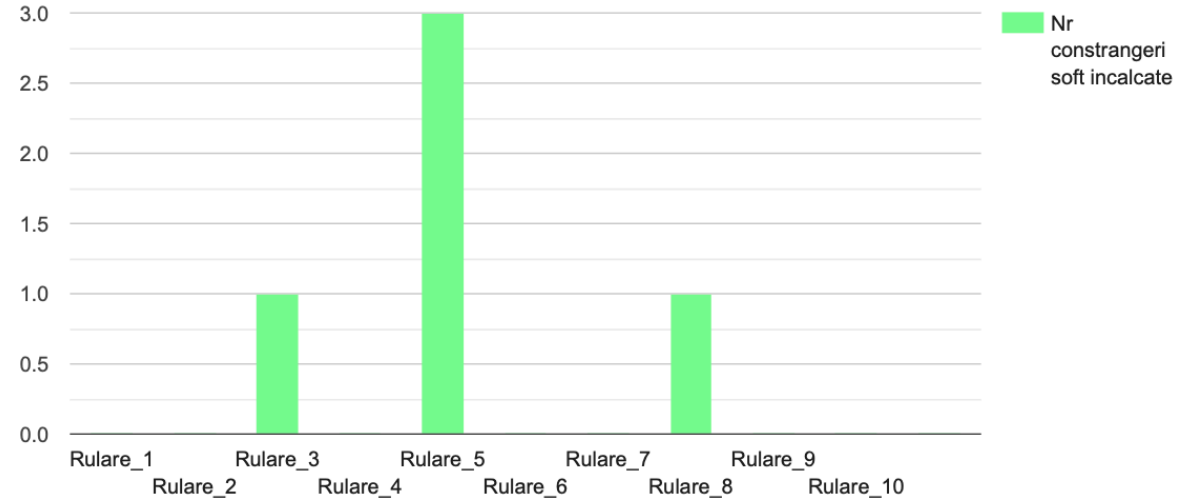
Orar_mic_exact (HC)

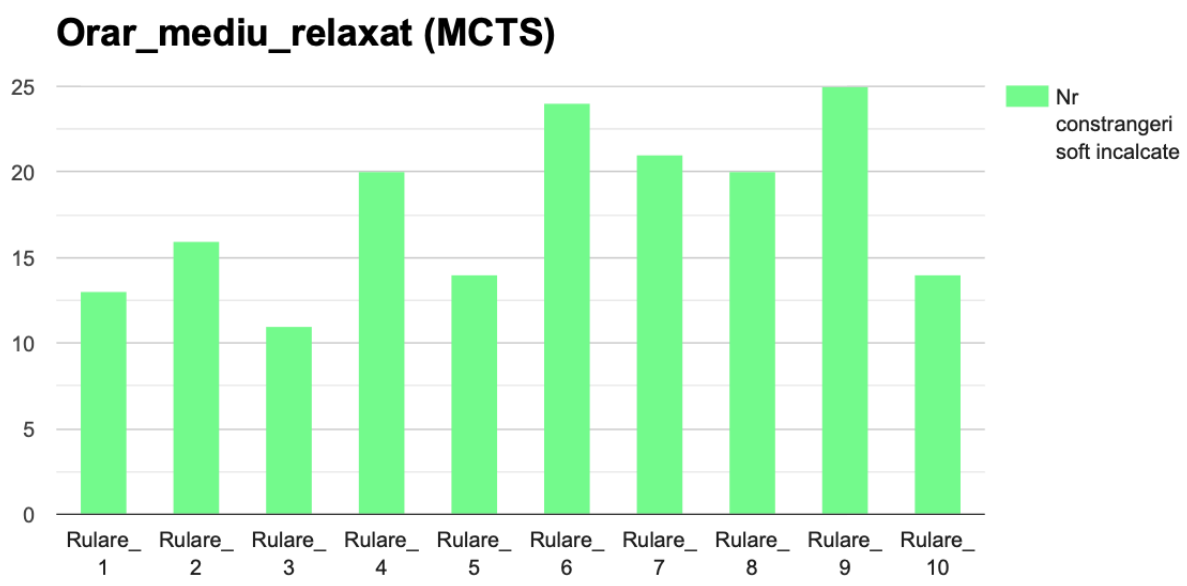


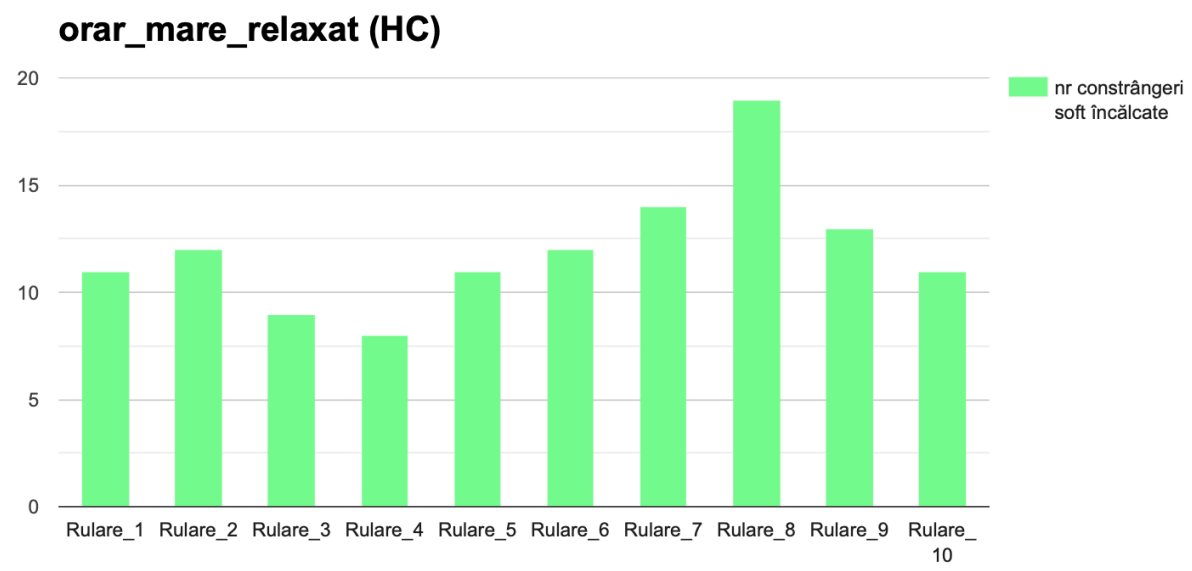
Orar_mic_exact (MCTS)

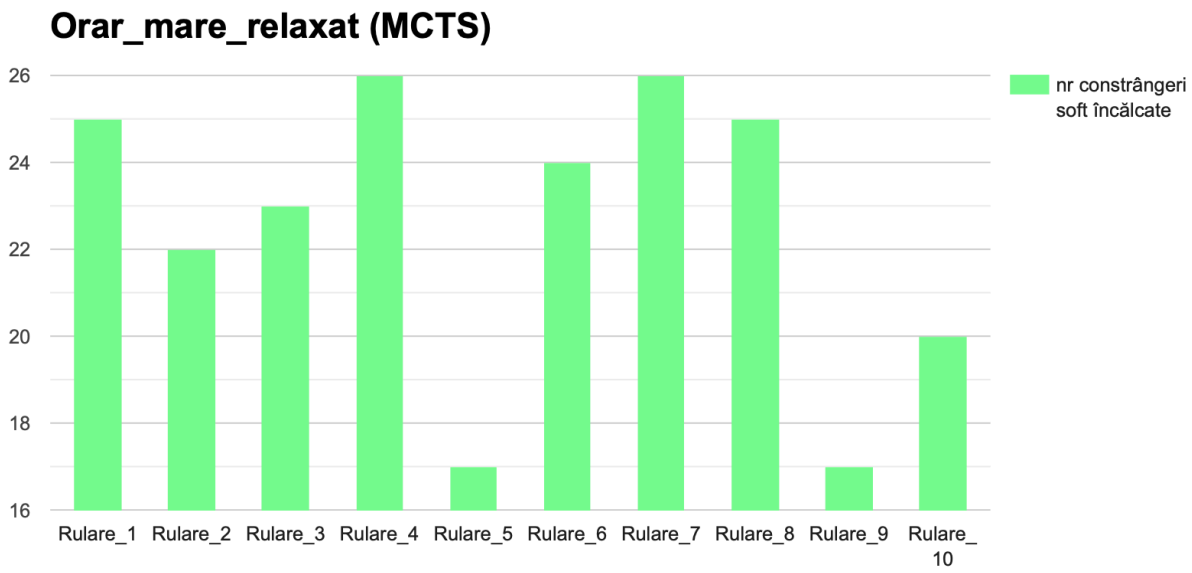


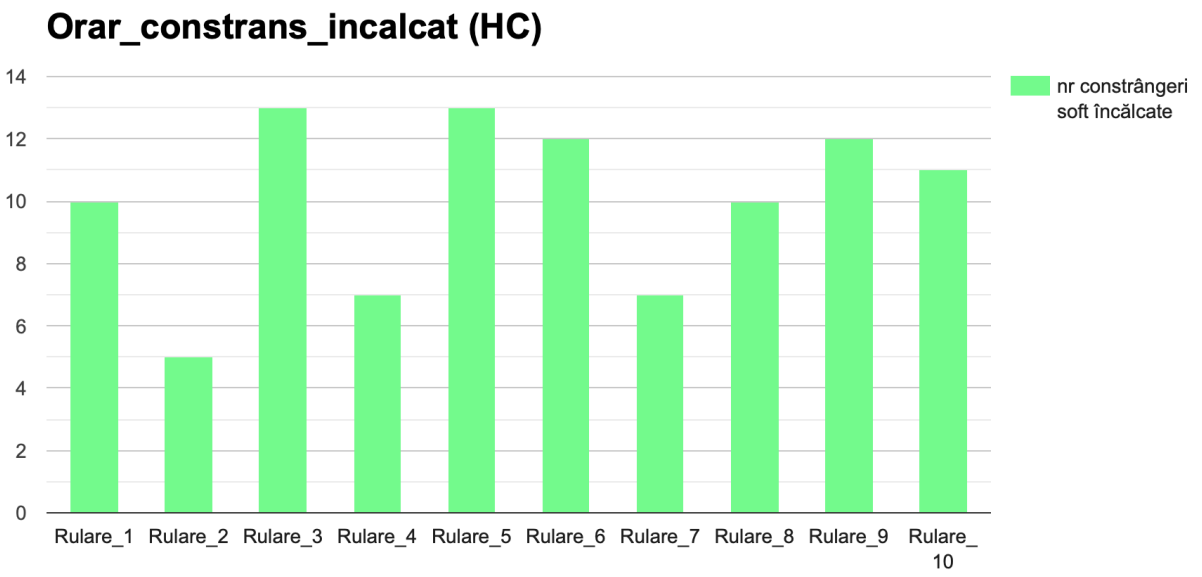
Orar_mediu_relaxat (HC)

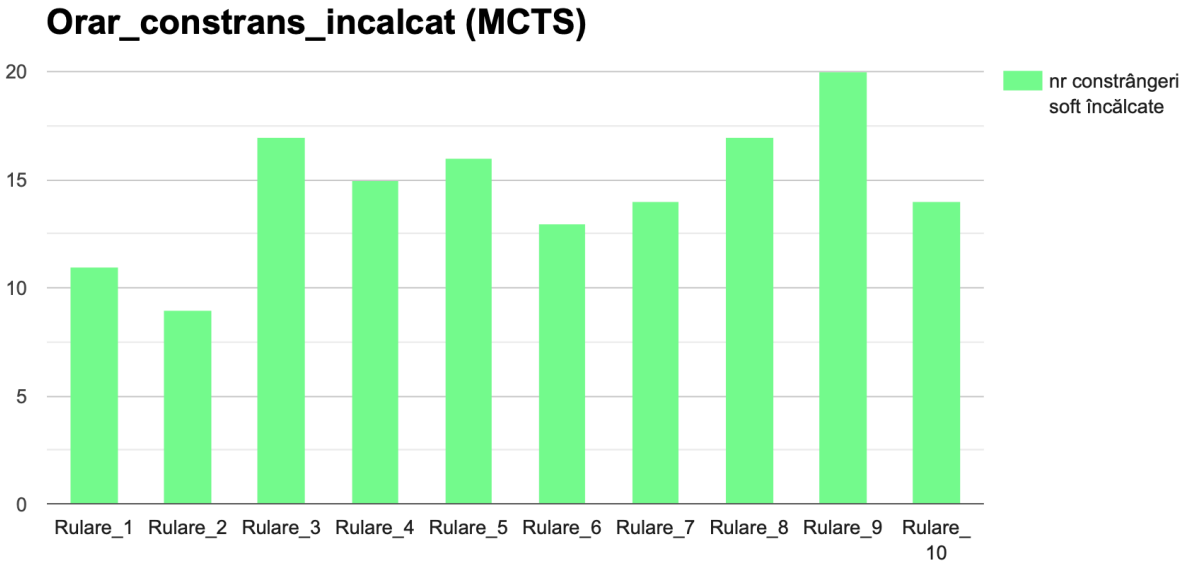



Orar_mediu_relaxat (MCTS)
25
20
15
10
5
0
Rulare_ 1
Rulare_ 2
Rulare_ 3
Rulare_ 4
Rulare_ 5
Rulare_ 6
Rulare_ 7
Rulare_ 8
Rulare_ 9
Rulare_ 10
Nr constrangeri soft incalcate


orar_mare_relaxat (HC)
20
15
10
5
0
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
nr constrângeri soft încălcate

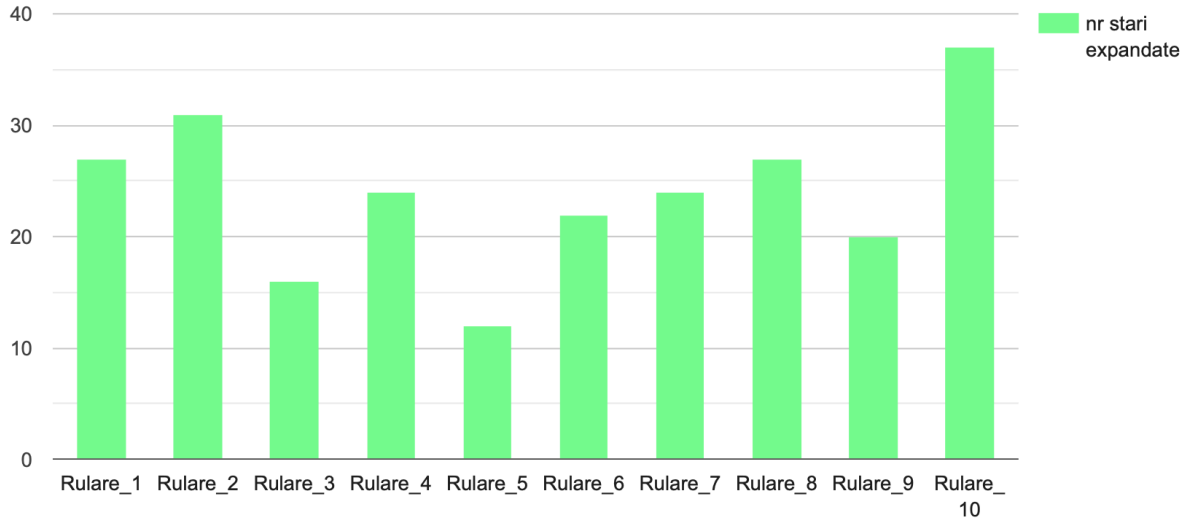

Orar_mare_relaxat (MCTS)
26
24
22
20
18
16
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
nr constrângeri soft încălcate



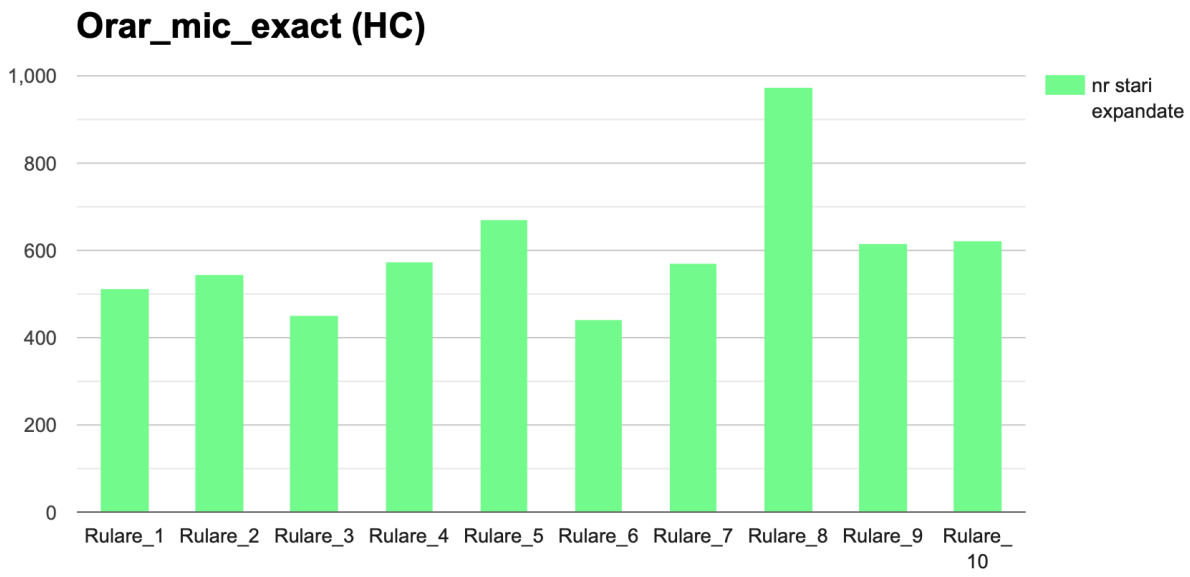


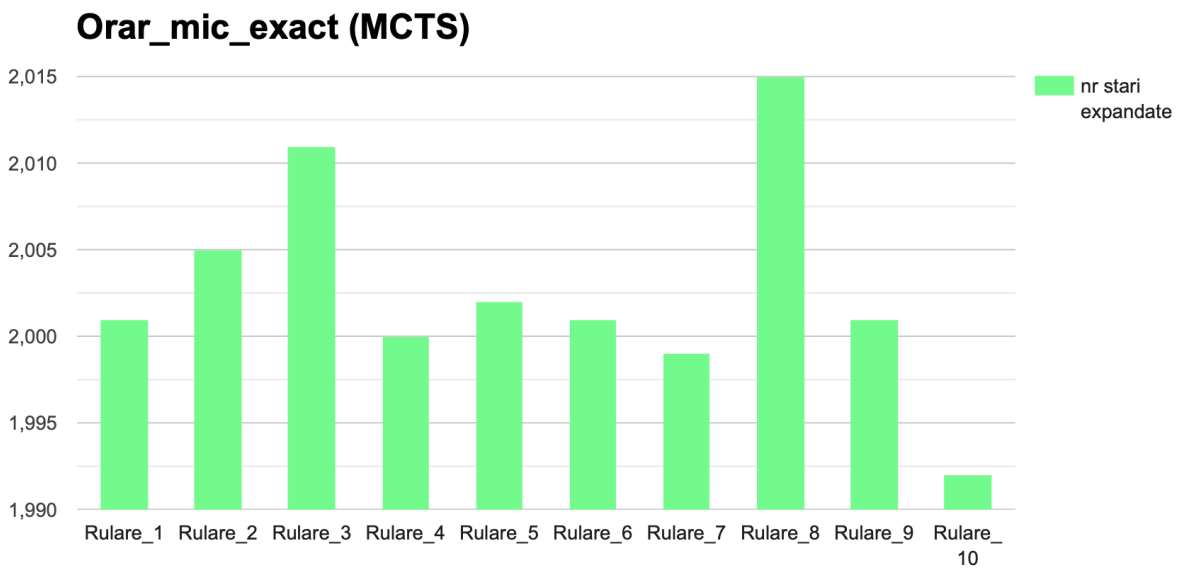
2. Numărul de stări expandate

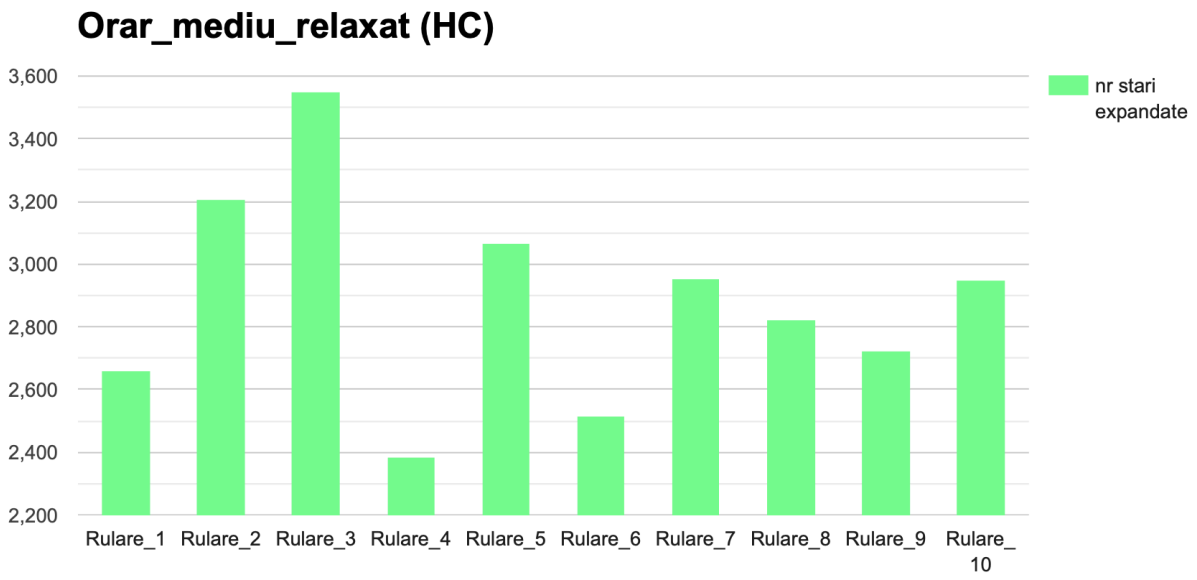
Dummy (HC)

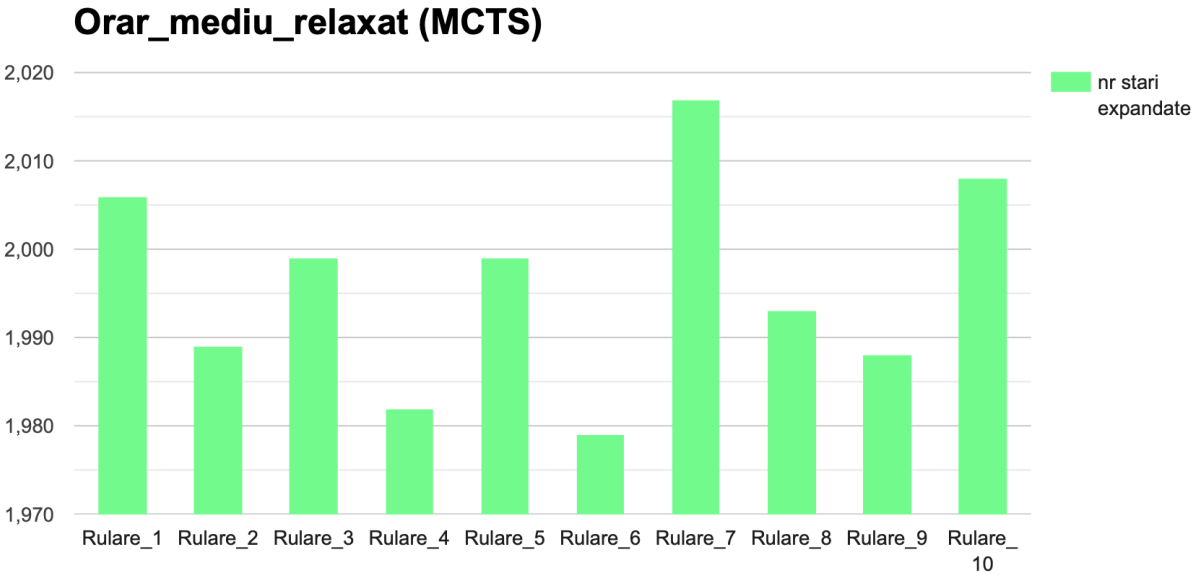


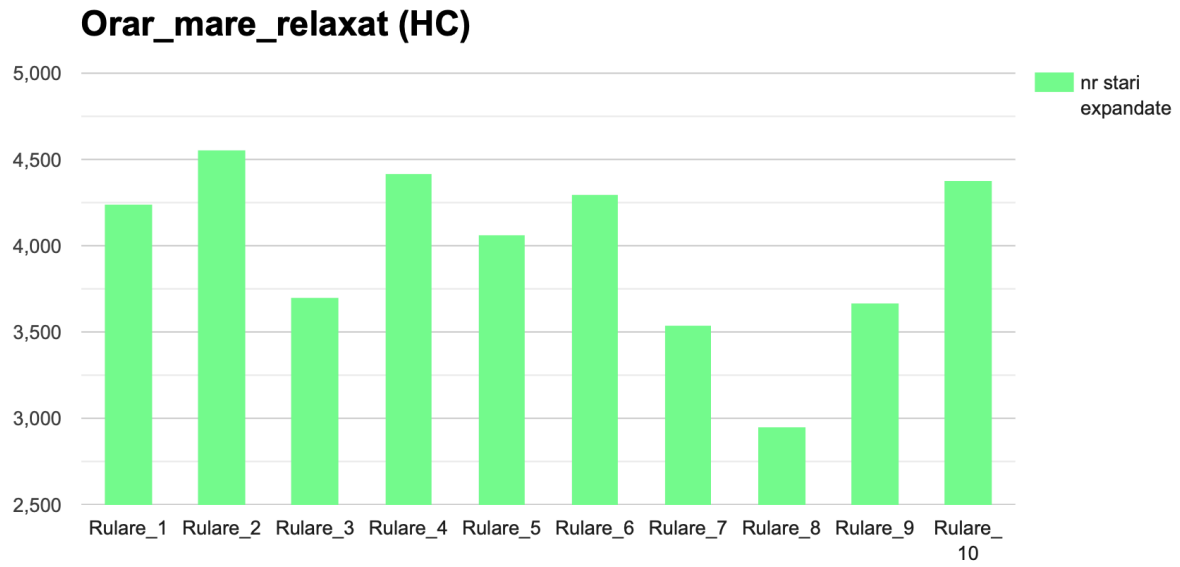
Dummy (MCTS)

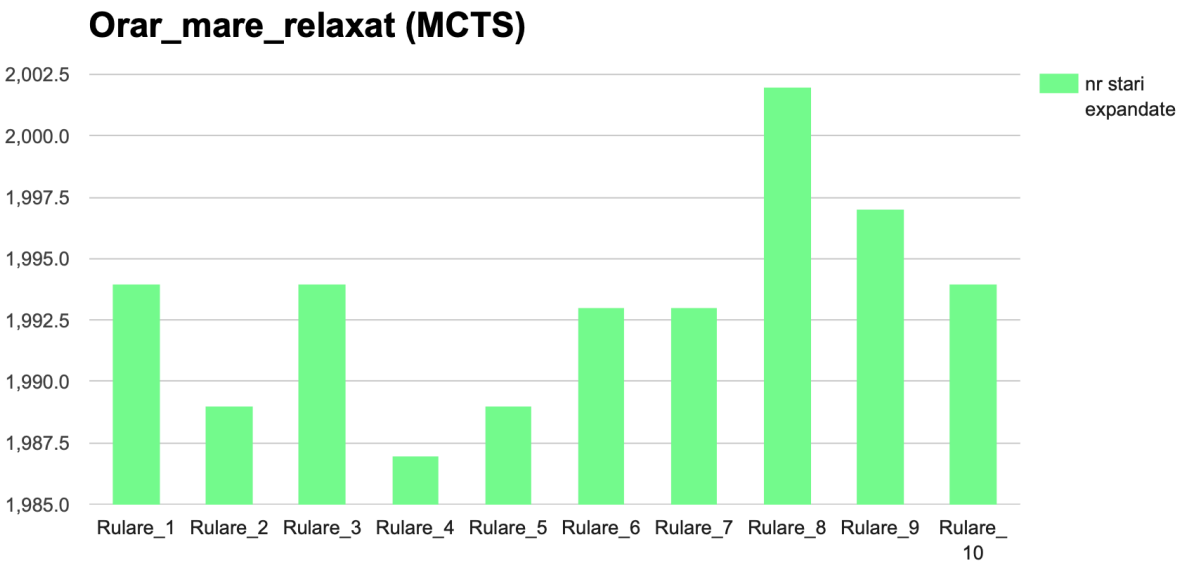

Orar_mic_exact (HC)
1,000
800
600
400
200
0
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
nr stari expandate

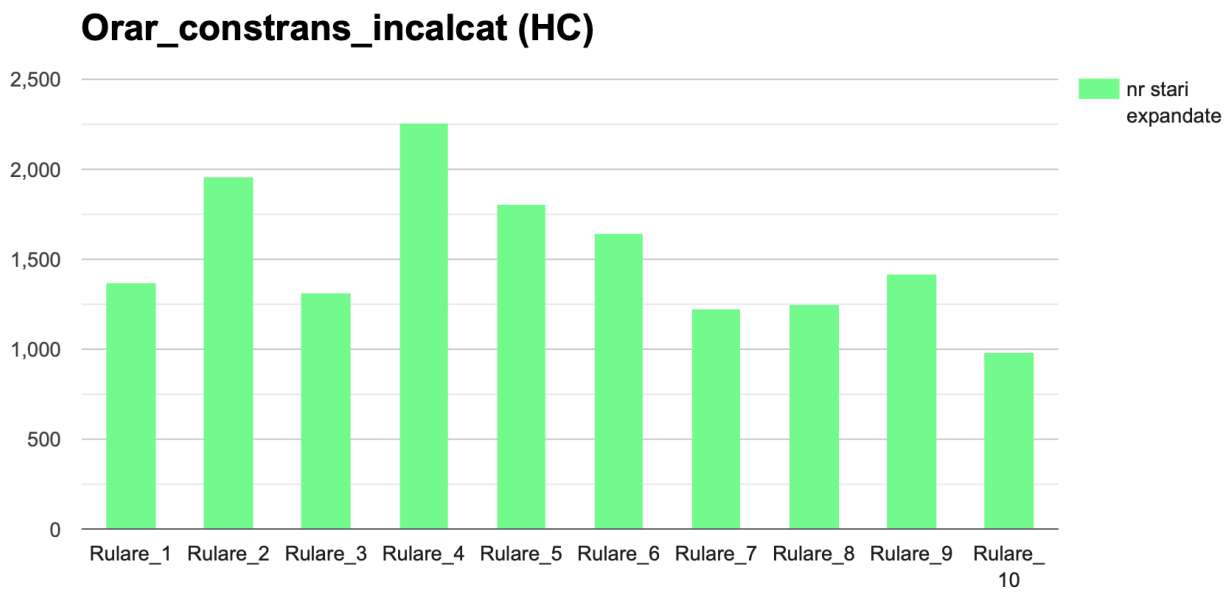

Orar_mic_exact (MCTS)
2,015
2,010
2,005
2,000
1,995
1,990
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
nr stari expandate

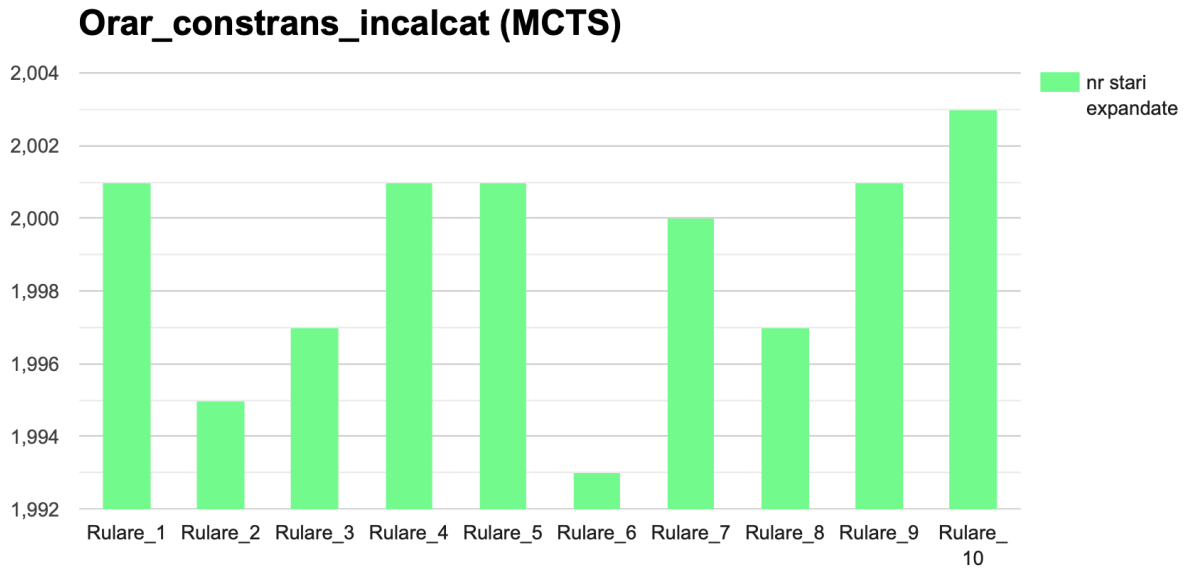

Orar_mediu_relaxat (HC)
3,600
3,400
3,200
3,000
2,800
2,600
2,400
2,200
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
nr stari expandate


Orar_mediu_relaxat (MCTS)
2,020
2,010
2,000
1,990
1,980
1,970
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
nr stari expandate


Orar_mare_relaxat (HC)
5,000
4,500
4,000
3,500
3,000
2,500
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
nr stari expandate

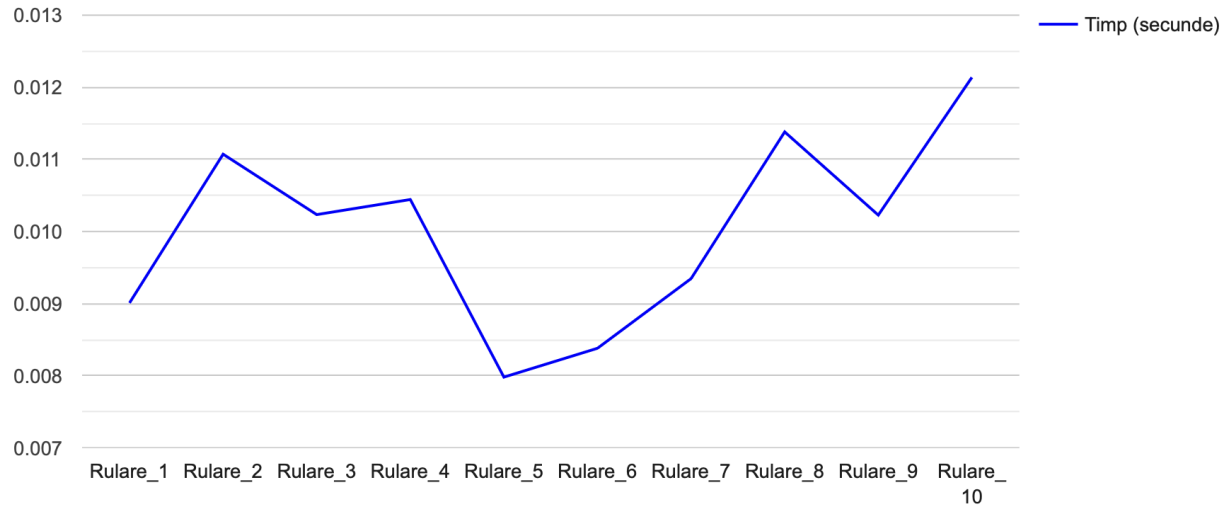

Orar_mare_relaxat (MCTS)
2,002.5
2,000.0
1,997.5
1,995.0
1,992.5
1,990.0
1,987.5
1,985.0
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
nr stari expandate



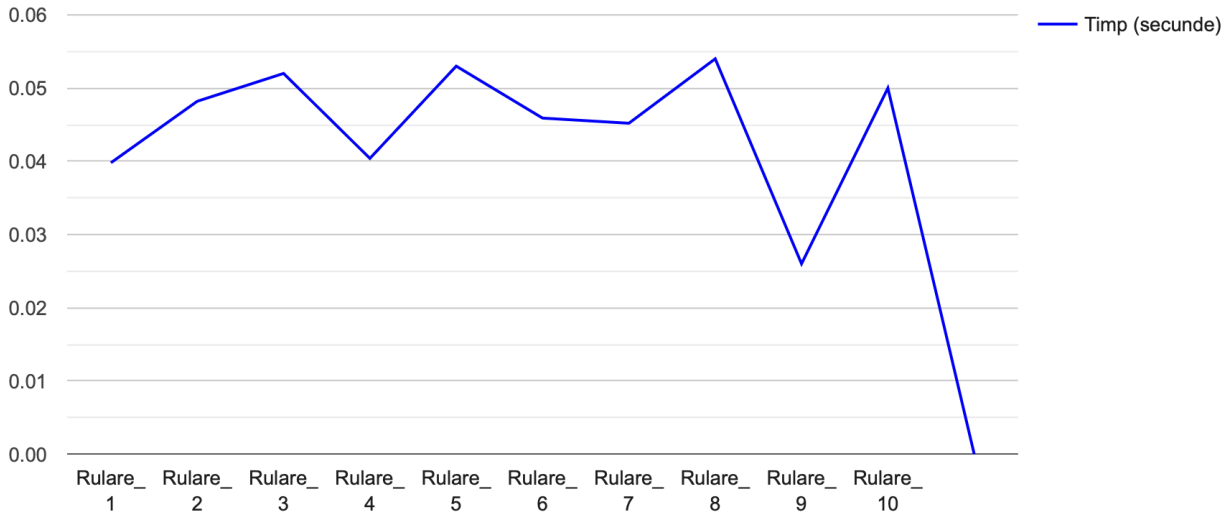


3. Timpul de execuție

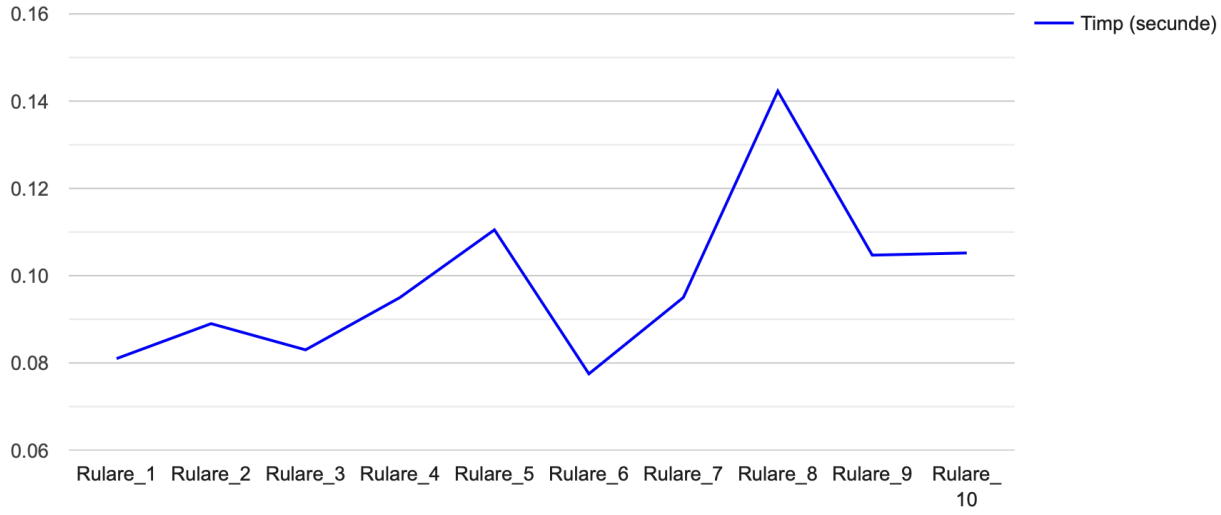
Dummy (HC)



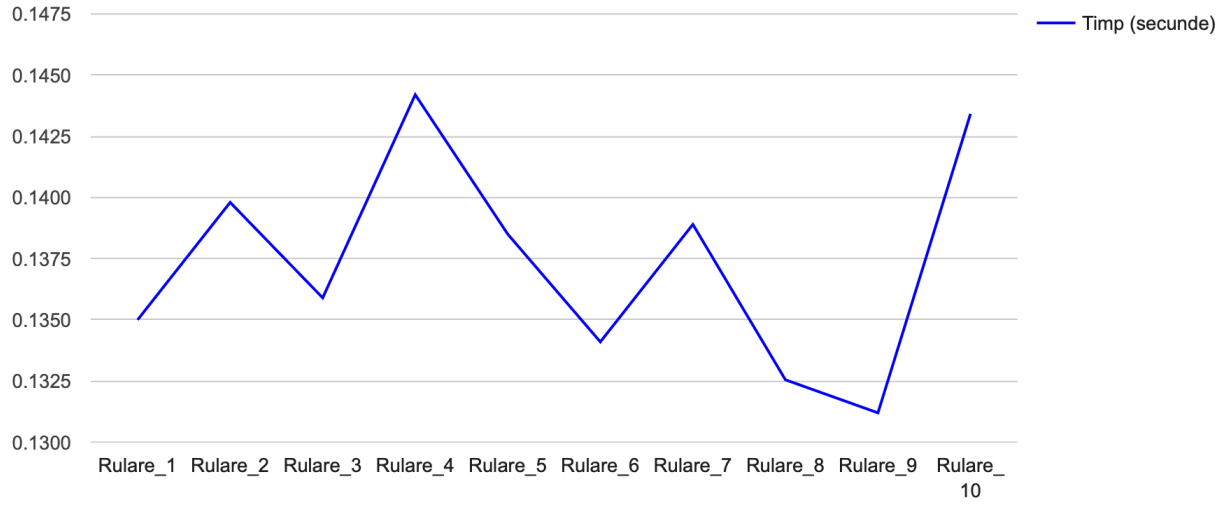
Dummy (MCTS)

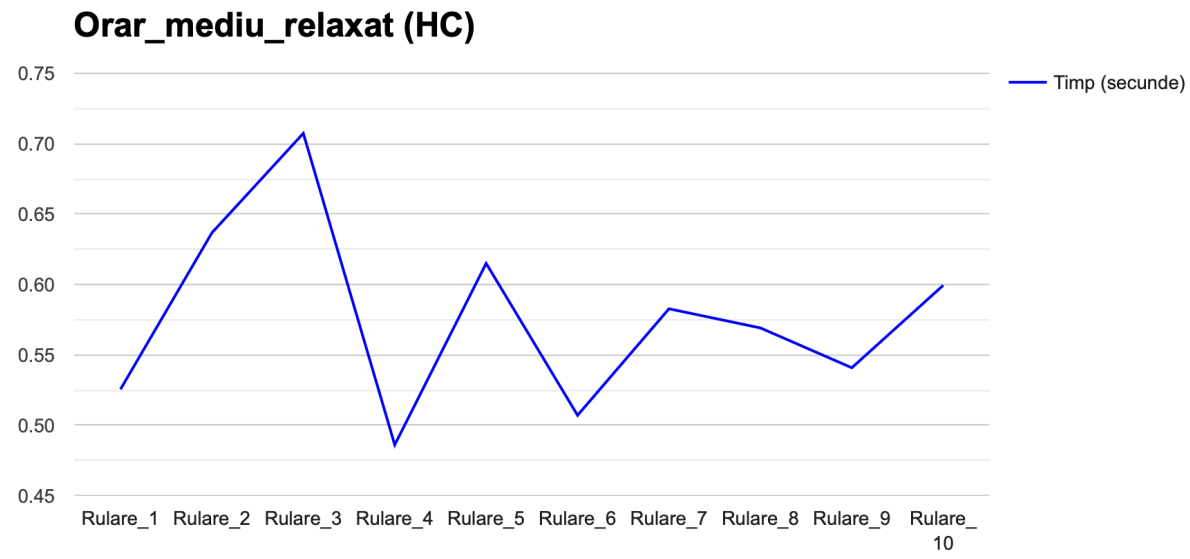


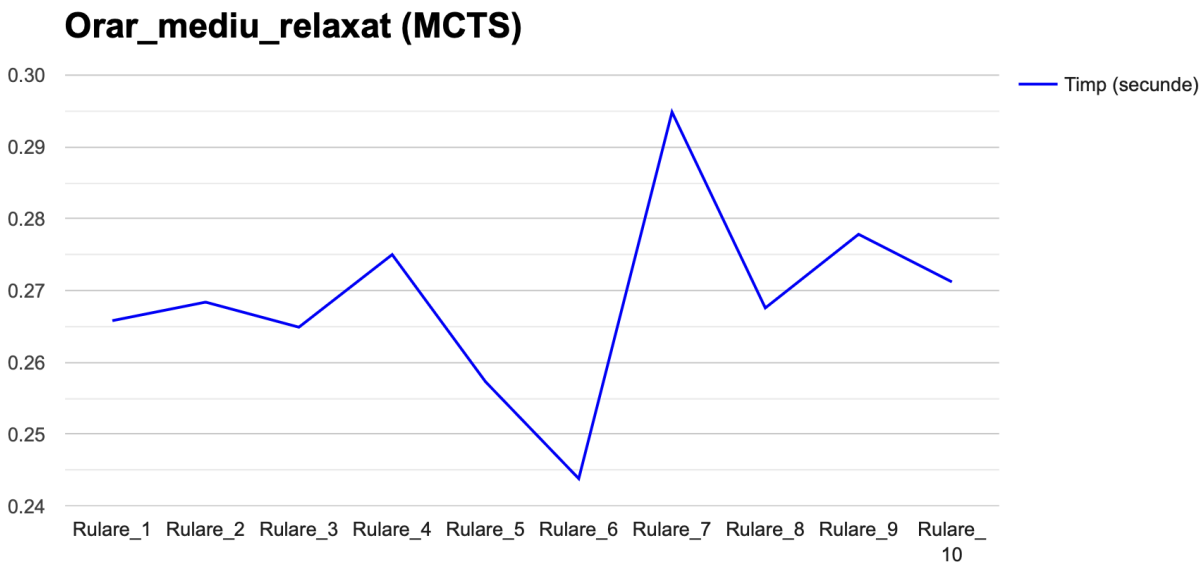
Orar_mic_exact (HC)



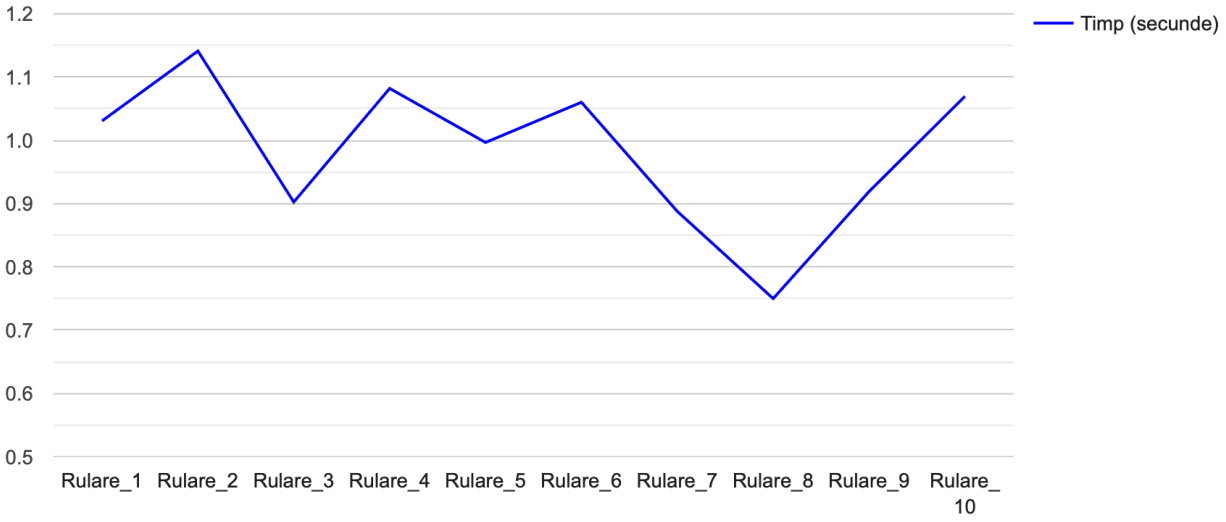
Orar_mic_exact (MCTS)



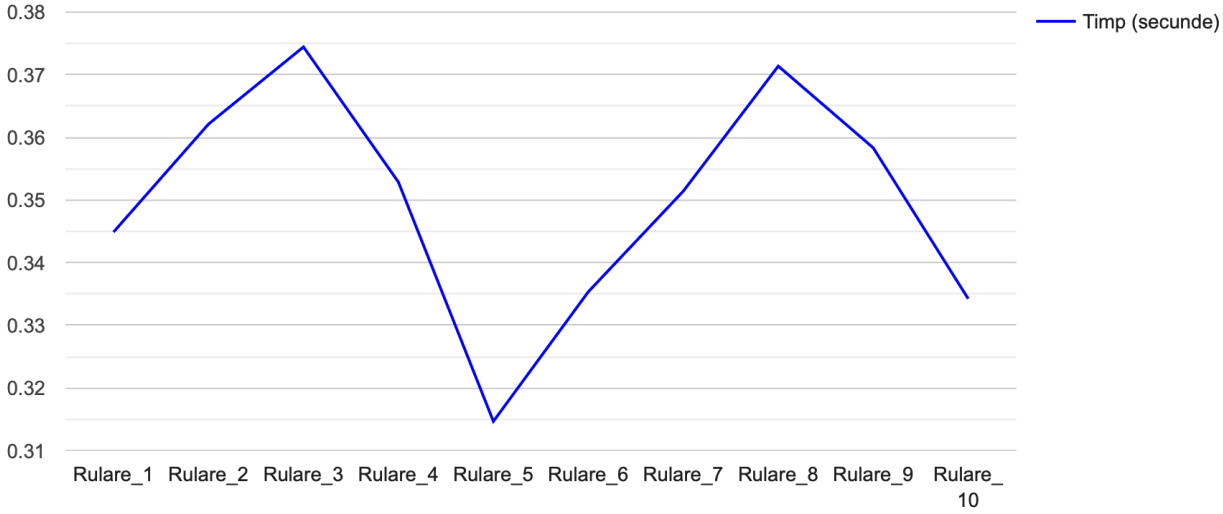

Orar_mediu_relaxat (HC)
0.75
0.70
0.65
0.60
0.55
0.50
0.45
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
Timp (secunde)

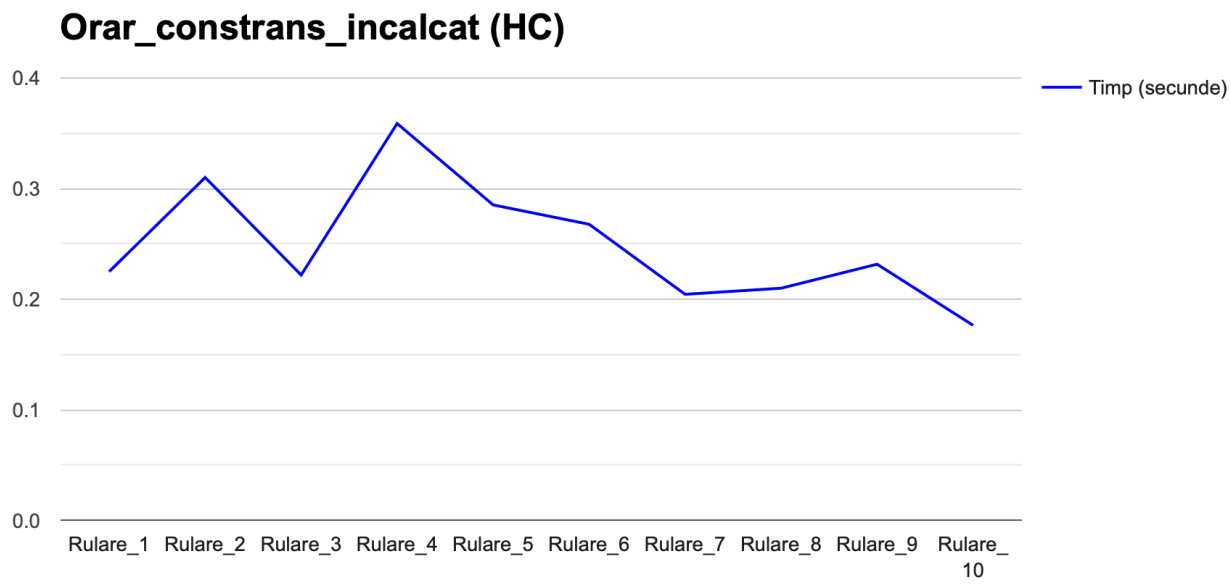

Orar_mediu_relaxat (MCTS)
0.30
0.29
0.28
0.27
0.26
0.25
0.24
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
Timp (secunde)

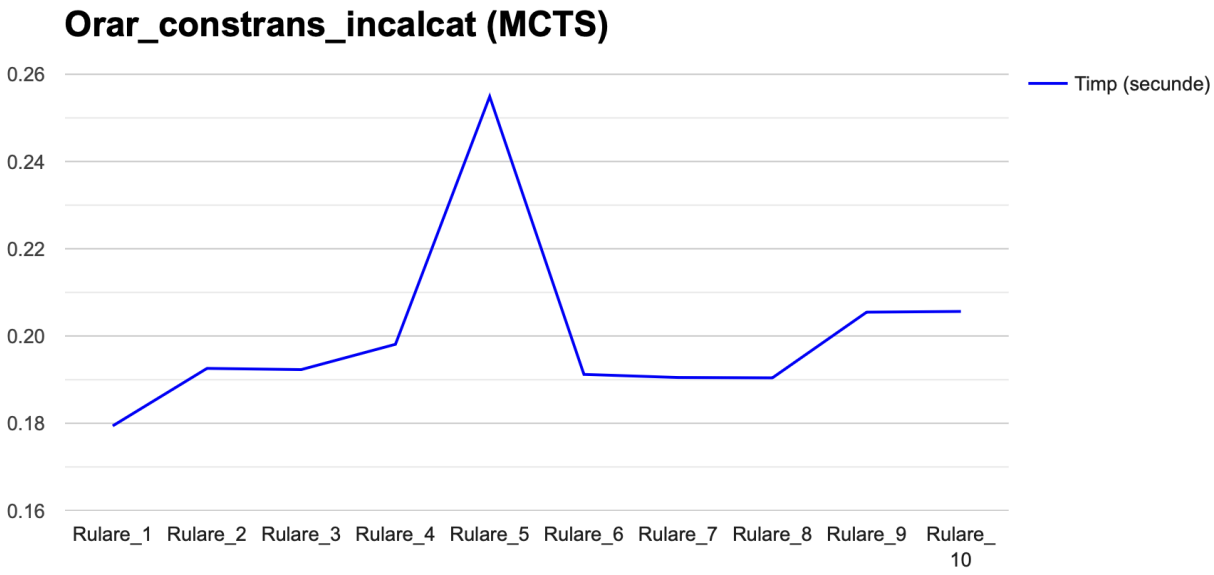
Orar_mare_relaxat (HC)



Orar_mare_relaxat (MCTS)




Orar_constrans_incalcat (HC)
0.4
0.3
0.2
0.1
0.0
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
Timp (secunde)


Orar_constrans_incalcat (MCTS)
0.26
0.24
0.22
0.20
0.18
0.16
Rulare_1
Rulare_2
Rulare_3
Rulare_4
Rulare_5
Rulare_6
Rulare_7
Rulare_8
Rulare_9
Rulare_10
Timp (secunde)