Exercise 5
Same Game
Artificial Intelligence for Games

Hubert Obrzut

December 29, 2020

CodinGame handle: **nan0S_**

1 Project Proposal

Project proposal was already sent by me on SKOS - I am in group with Szymon Kosakowski and Kacper Szufnarowski.

2 SameGame Easy

In order to get at least 6500 points from the SameGame puzzle I have just written random agent - pick any legal action and perform it.

3 SameGame Medium

In order to get at least 30000 points from the SameGame puzzle I have written BeamSearch algorithm with no additional enhancements - $50ms$ for every round (including the first one despite $20s$ limit, search did not use any state). To be more exact, I got something around 30500 points. I have used beam width equal to 100.

4 Zobrist Hashing

With state hashing I have achieved score of about 38000 (37900 to be exact) - every state was hashed depending on the position of individual tiles. Better score is understandable as with the SameGame, we can get to one state with a different sequence of moves - especially on some levels and especially at the beginning, which could lead to duplicate states in our search, which will lead to worse search effectiveness.

5 Selective Policy

With color taboo policy I have managed to achieved score about 50700 - I have just added punishment to the current score, if I have decided to press on the most frequent color. This improved the score by aggregating the most frequent color into larger blocks and pressing on then only at the end.

6 Nested Monte Carlo Search

Nested Monte Carlo Search is very costly, but it has proved to be able to find very good solutions. After a few experiments I have decided to run NMCS with depth equal to 2. I have written NMCS to run in the time constrained manner - I have used $20s$ limit in the first round and $50ms$ in the rest of

the rounds (running NMCS for $50ms$ improved just a little bit, $20s$ in the first round was the main part). With pure NMCS I have managed to get the score around 66000. When I have added color taboo policy I got 107000 points - when calculating possible moves, I did not consider moves removing the most frequent color as long as a had other moves to do. As we can see this had a major impact on the efficiency of the algorithm. I wanted to deepen the search to 3, but unfortunately $20s$ was too little time to find meaningful solutions - although better solutions were found, which was not the case with Beam Search - extending time limit did not have such significant impact on the quality of solutions.