

- a. I ran the alpha-beta agent with different depths. I have a maxDepth variable to control the search depth of the agent. By default I have it at 6. But I will try a few others to see the results and compare them.

Depth	Did it improve the moves?	Did it make it worse?	Did it make the game slower?
1	No	yes	no
4	No	yes	no
8	No	No	yes
12	yes	No	yes

So increasing the search depth slows down the time per move and improves the moves the agent was making. Lowering it is the opposite.

For improving the evaluation function, I added a king bonus to my implementation to make the agent value the king higher than normal pieces. This avoids the loss of kings instead of treating them as normal pieces, which in turn improves the evaluation. Improving the evaluation allows for the agent to improve at playing.

- b. When using $C = \sqrt{2}$ I won against the agent, but it played very well, and when I set it to $C=2$, it tried a lot of moves that I hadn't seen it do before, and beat me
- c. When playing alpha-beta, it performed significantly better than the Monte Carlo agent. The alpha-beta did a lot better when it came to setting up jumps than the Monte Carlo agent. When trying out the hybrid agent, it was a good mix and was probably the best I played when testing the agents.