

Robert Schneider
Artificial Intelligence
Planning Agent

This planning agent implements the same functions as SearchAgent.java. It uses the same techniques to explore the map, and store the data. To learn more about how that works, please refer to rschneiderP2.report. The function Explore in this agent, also implements the ability to open doors and break rocks. If, at any point the agent sees a rock or door, and knows where a key or hammer is, it will go and try to open that door/break that rock.

I implemented my iterative deepening version of depth first search to keep track of already visited nodes, to help with time complexity, however it still takes a while as the depth goes to around 20. I thought that maybe it would be effective to implement a version of bi-directional iterative deepening depth first search, but did not get to it. This would cut the time complexity to about half, but the problem still lies on big maps. I modified the search algorithm to be able to choose if search ignores rocks and doors or not. This is helpful because if you call search twice, once ignoring and the other time not ignoring. As a result, you could know that there is an obstacle in your way. However this would only be effective on maps where the cheese is located in a small room blocked by a door.

I also implemented the Planning Agent to work on maps that are larger than the normal size, however to do this, you must go into the code, and change the variables gridSize, and gridYSize. These values need to be set to a value that is > than double the actual grid size. These variables are at line 28 and 29, initially set to 60 and 40.

Although the planning portion is not the most optimal, it does solve a good amount of worlds. The planning portion will solve obstacles at any point if it is able to, and then continue to explore once its finished. I did find that there are certain bugs in my code, that break on certain maps, but I can not quite figure out why, for example, my code breaks on the map worlds/tw22, as well as some other maps. What will happen is my search algorithm will go into infinite search without actually searching, just printing out the depth, and I don't know why its not searching. It works most of the time, but every once and a while, the agent will be in a certain orientation where it hits this bug and breaks.

In order to add more objects to this Planning Agent, consider adding things to the following lines, and look at comments to see what to do exactly. You must know the string representation of the object you wish to add, as well as the integer values of those string (important for the buffered reader when processing retinal fields). See code for more details.

47-49

300-326

370-374

517-520 <need to know the int value of the strings implemented>

667-670