Mobile robot exploration

When implementing the random agent I decided to make the robot spin 0, 90, 180 or 270 degrees and then giving it a random direction to walk (either backward or forward). The robot will continue walking this path for a while before getting a new random turn and direction.

In the fixed agent I used trial and error to figure out a specific path that would let the robot pick up some of the static energy blocks. I did not make it pick up all constant blocks since this process was time consuming.

When designing the reflex agent I gave it the ability to notice when it was close to a wall using its sensors. When it gets closer than a certain distance to a wall it turns either left or right depending on which sensor that has the closest readings. If both sensors are within a close range of a wall then it means the robot are directly facing a wall and needs to back off and turn before it can continue. When none of the sensors are close to a block, the robot just walks forward. If a block gets within a certain range, the robot adjusts its direction against the block.

In the memory agent I also added information about if it has collected a block within a certain time period. If it failed to do so it means the robot is stuck somewhere or are following a bad path. If this is the case then the robot will enter a state where it just avoids walls but does not search for blocks. After some time in this state, the robot will continue its search.

Poker game agent

I implemented the poker agent to perform its betting by obtaining a random number between 0-50 and placing the bet into the game pot.

I gave the fixed agent instructions to perform a different bet depending on what round it is. For the reflex agent i created a point system where a pair would yield 10k points three of a kind would give 20k. Depending on what cards the player had obtained its par or TOAK in the score would increase with 100 times the value of the card. Lastly the player would get a bonus based on the sum of the values of the hand. I used this exact same method to compare who wins the game. The reflex agent, as expected, performed much better than both the random and the fixed agent. It wins every single time I run the program.

I also started the implementation of the memory agent but unfortunately i did not have the time to finish it completely.

The different agent implementations can be found in the bet() method where different betting procedures is defined based on the players type.