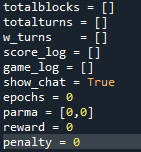
Kevin Charles Hostler

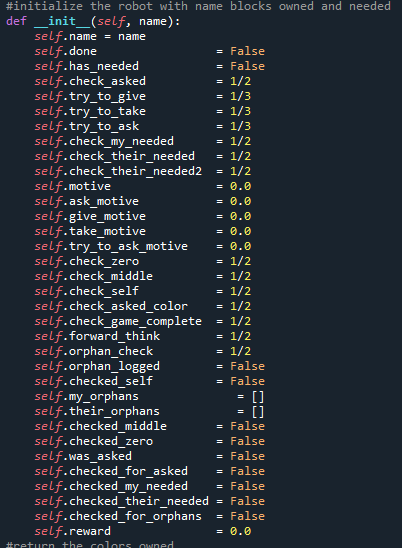
Experimental phase update Version 2:

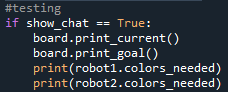
This report is to document and better help explain the plots shown in the PowerPoint slides. To start with we have two different versions of the logic code, though both have a similar premise, but the second version implements fixes to the issues found out in version one.

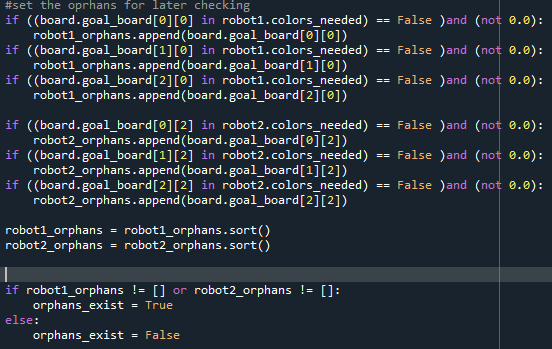
**Version 2**

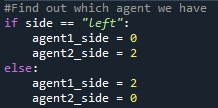
This logic version was created and edited with several key ideas in mind. To start with version 1 would end up with the agents getting stuck for several turns when blocks would fill up the middle or they would get the blocks they need but leave blocks neither need but one needs to take in the center. So this boils down to the version 1 agents not being able to take actions that have future rewards as in completing the game before turn time-out.

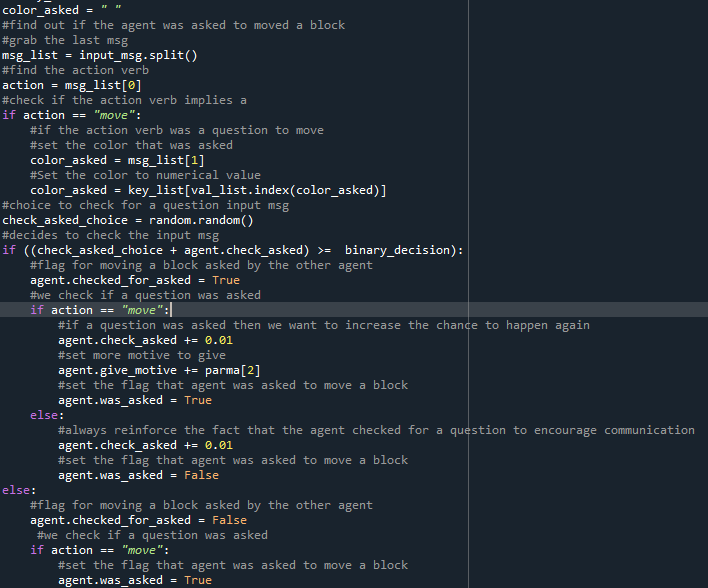
The first thing we do is set all of the variables that are tracked each game to global to be used for the graphs at the end.

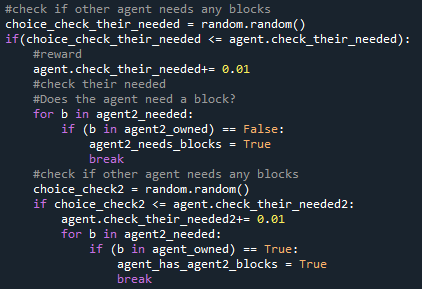
 The new additions to each agent are the decision chances for forward thinking checking, orphan block checking, game completion checking and orphan logging.

 The second major change to help the optimization time of training was limiting the i/o of the print statements and have moved all communicating to internal lists, though for testing the messages can be printed again by a simple Boolean switch.

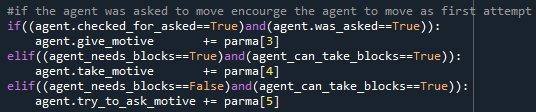
 The next major change is the addition of an orphan verification system where before the agent and game loops begin we verify which blocks are never needed and if they need to be moved because of it. This will be used by agents during decision making.

Set the agent coordinates so that we don’t do this in the decision loop.

 The check for a question logic is mostly the same, though this time the rewards have change and can be altered in loops for testing. The agent also keeps better track of what occurred for further decision help.

The next major change was to have the agent decide to check if the other agent needed any blocks outside of the decision loop. This greatly helped to reduce the training time.

 This new logic does a great deal of heavy lifting as it is the forward thinking decision. This has the agent decide if it wants to check for possible ways to end the game. It finds out if it doesn’t need to take blocks but should, then what those blocks that aren’t needed but should be moved, and where who they should belong to.



After the game completion check, the agent gets motive increase for it’s first action based on what was found out.