**COSC 4368 Group Project**

*“Using Reinforcement Learning to Discover Paths in a Transportation World”*

**Introduction**

For this assignment our group designed an agent-based system which over several iterations explores and learns an initially unknown environment, a five by five grid consisting of three different types of spaces, normal locations, pickup locations, and drop off locations. This agent was tasked with moving items from the aforementioned pickup locations to the aforementioned drop off locations, with the aim that over time, through reinforcement learning the agent would become more efficient with their movements and develop the capability to complete their assigned task in fewer iterations. Our group conducted five different experiments, each using a different combination of policies in order to better understand their effects on the agent and its ability to solve the task at hand. Each time an experiment is conducted, data we have deemed interesting, Q tables, the number of times each cell is visited, the percentage of cells visited, etc., is outputted to a file. Initially, we treated our Q Tables slightly differently from what we have seen thus far, instead of simply considering the Q values of each action of each cell, we also took into consideration the Q values of each action of each cell of each permutation of the location, whether or not the agent was holding a block, and the states of each of the pickup and drop off locations, as in whether or not blocks were available to pickup from pickup locations and whether or not drop off locations were full. Albeit interesting, this resulted in an overwhelming amount of data to comb through, thus in addition to these Q Tables, we also produced Q Tables simply representing the Q values of each action of each cell, while taking into consideration whether or not the agent was holding a block. This paper will serve as a medium for our group to communicate the findings of each run of each experiment, any interesting trends we may have noticed in each of these experiments, and which combination of learning algorithm and policy type, in our opinion based on our results, proved to be the most useful.

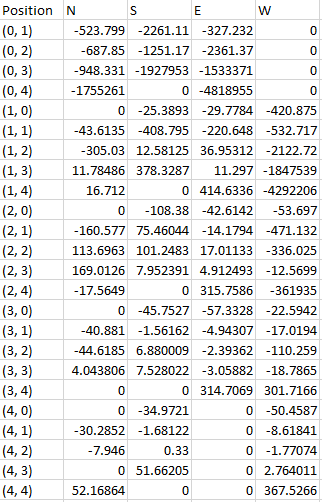
**Experiments**

As previously explained, we have conducted two variations of each experiment, of which there are five, twice in order to accurately ascertain the efficacy of our agent and learning algorithms. Over the course of these experiments we implemented three different policies to influence the agent’s chosen actions, PRANDOM, which causes the agent to choose an action randomly if the pickup or drop off actions are not available, PEXPLOIT, which causes the agent to choose the action with the highest Q value if the pickup or drop off actions are not available, and PGREEDY, which is similar to PEXPLOIT where it causes the agent to choose the action with the highest Q value if the pickup or drop off actions are not available, but handles ties between Q values differently. As previously mentioned our group produced Q Tables for each permutation of states, but due to the sheer number of Q Tables and limited number of space we will not be able to display them within this paper, but we will attach .csv files containing our data as well as detailing briefly their contents and the corresponding heatmap displaying the number of visits to each cell. We will instead include Q Tables corresponding to our reduced state space, of which their will be screen shots. The pickup locations are located in cells (0,0), (2,2), and (4,4) while the drop off locations are located in cells (4,0), (4,2), and (1,4).

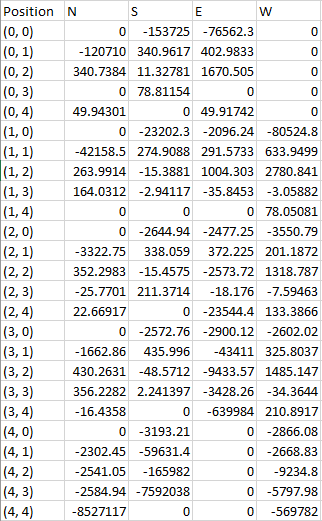
**Experiment 1**

For the first experiment our group used an αvalue of 0.3 and a γvalue of 0.5. Using these values, we applied our implementation of the Q-Learning algorithm for 4000 steps using the PRANDOM policy, followed by 4000 steps using the PGREEDY policy.

**Experiment 1: First Run**



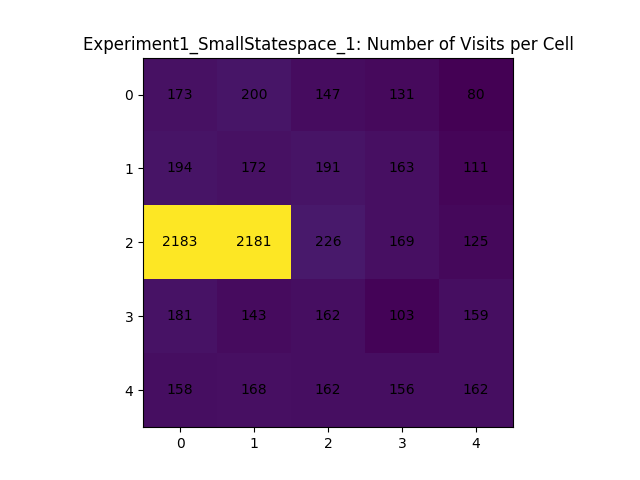
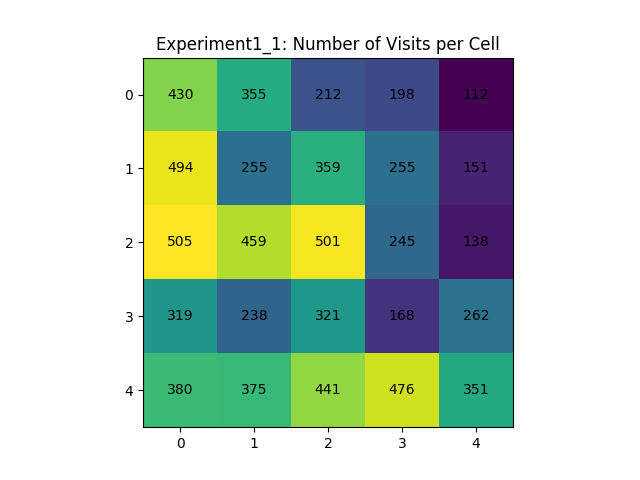
*While Agent is not holding a block*



*While Agent is holding a block*

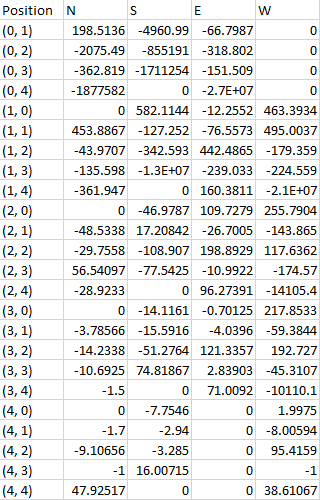
As can be gathered from the above the Q Table on the left, cells containing a pickup location or cells near a cell that contains a pickup location have higher Q values, as this Q Table represents when the agent is holding a block, as is evidenced by the Q values around cell (0,0). If you view cell (0,1)’s Q values for its actions, it is apparent that the agent is not at all incentivized to move in the direction of the pickup location (as it is already holding a block) and is instead incentivized to move East or South, in the direction of drop off locations.

Similarly, the Q Table on the right, which corresponding the Q values of each action of each cell while the agent is not holding a block, actions which lead the agent to a cell containing a pickup location are higher than those which do not.

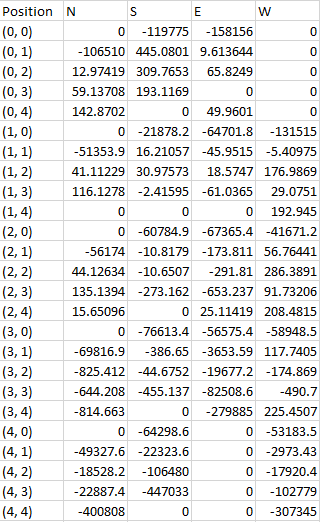
The above heatmap represents the number of visits per cell, sourced from the same data as the above Q Tables. It is extremely interesting that the agent seemed to have alternated between cells (2,0) and (2,1), as well as visiting them so much more frequently the rest of the world. One assumption is that those cells are visited so frequently due to their placement between a pickup location at (0,0) and a drop off location at (4,0), as well as the pickup location at (2,2) and (4,2), however it does not explain why the other cells in that path are not as visited as well, thus it seems the agent was nigh stuck in a loop of sorts. This phenomenon could be a result of an incorrectly implemented algorithm or simply a poor combination of inputs into said algorithm.

Unlike the heatmap corresponding to the smaller state space as depicted above, the heatmap associated with our initial, fully fledged state space seems to be much more well evenly dispersed and the outcome of which aligns with expectations. As is shown in the figure above it is apparent that cells containing pickup locations or drop off locations, or are at least on a path between pickup and drop off locations, are much more frequently visited then parts of the world where there is less incentive to visit, as in the top right portion of the world. However, it seems as that even though there exists a drop off location at cell (1,4) it seems to have been overlooked due its surrounding cells having less incentive to visit, as all other locations have a much higher number of visits.

**Experiment 1: Second Run**

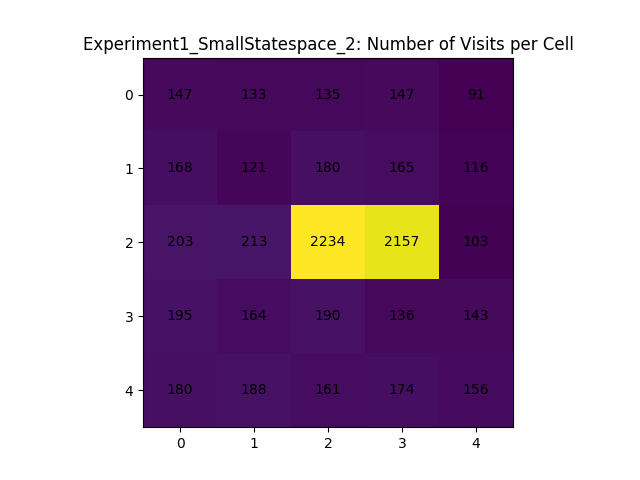


*While Agent is not holding a block*

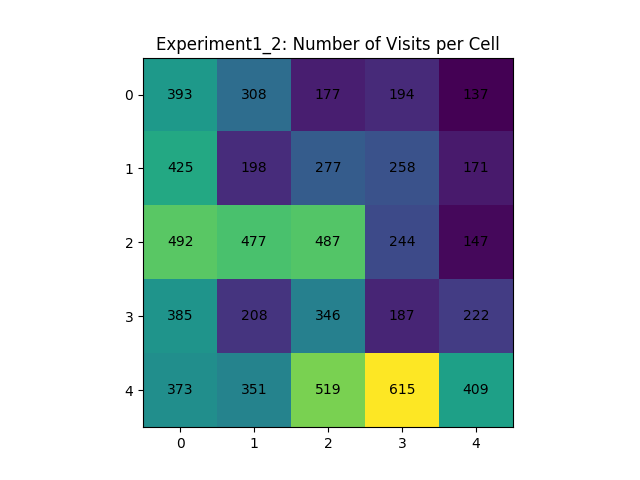


*While Agent is holding a block*

Compared to the first run, I find that these results are less representative of ideal agent movement in some instances while better in others, through the visualization of the Q values corresponding to cells at or around pickup and drop off locations. During the first run of experiment 1 the results were indicative of the same trend, however there existed idiosyncrasies, as is also the case here, such as our favorable locations being ignored. Here we can see that in the left Q Table, when the agent is holding a block while being located in cell (3,0), it should be incentivized to move south towards the drop off location one cell below in (4,0), however the Q value associated with that action is a largely negative values and as such is not the ideal reward we would be seeking. However, looking at this entry in (3,0) may explain the frequency of visits per cell as evidenced by the heatmap associated with the first run of experiment 1 and this the second run which is pictured below. All actions for this location are large negative values, except for the action corresponding to North, which is 0. These values indicate the reason behind the high frequency of visits to location (2,0) in the previous heatmap.



This heatmap is very much similar to the previous run of experiment, as is evidence above. The agent seems to have oscillated between two cells while disregarding everywhere else in the world.

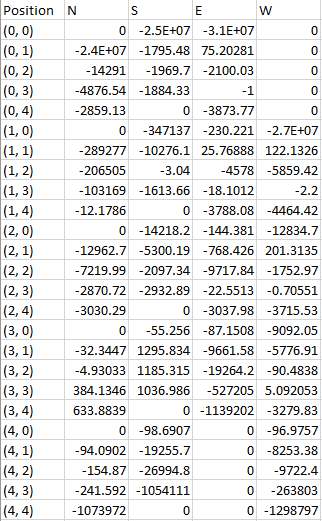


Similar to the first run of experiment 1, this heatmap is indicative of the results we would expect from a properly implemented learning algorithm, as the frequency of cells visited has nice, evenly distributed data. Interestingly, the agent seems to have taken a particular liking to the path between the pickup location in cell (4,4) and the drop off location in cell (4,2), which just so happens to be cell (4,3), the most frequently visited cell by the agent. However, this is also indicative of the agent taking these as favorable paths while disregarding the other targeted locations.

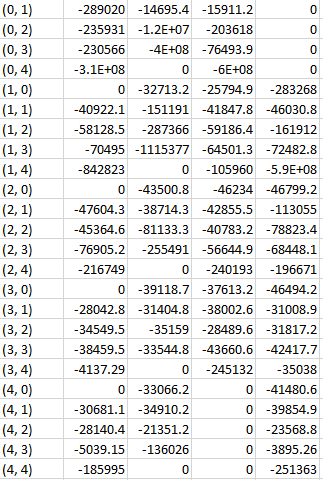
**Experiment 2**

For the second experiment our group used an αvalue of 0.3 and a γvalue of 0.5. Using these values, we applied our implementation of the Q-Learning algorithm for 200 steps using the PRANDOM policy, followed by 7800 steps using the PEXPLOIT policy.

**Experiment 2: First Run**

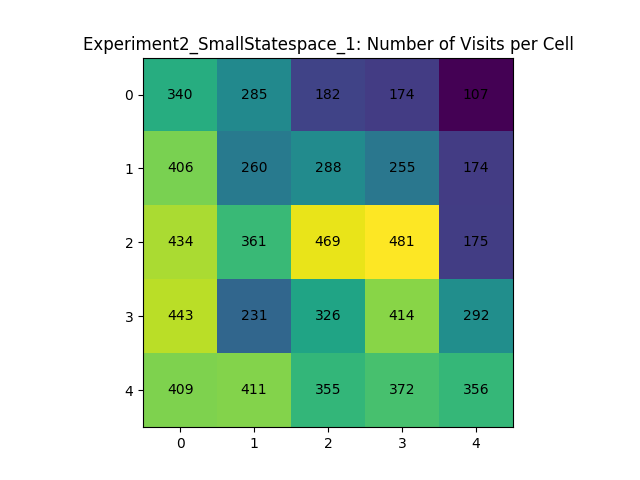


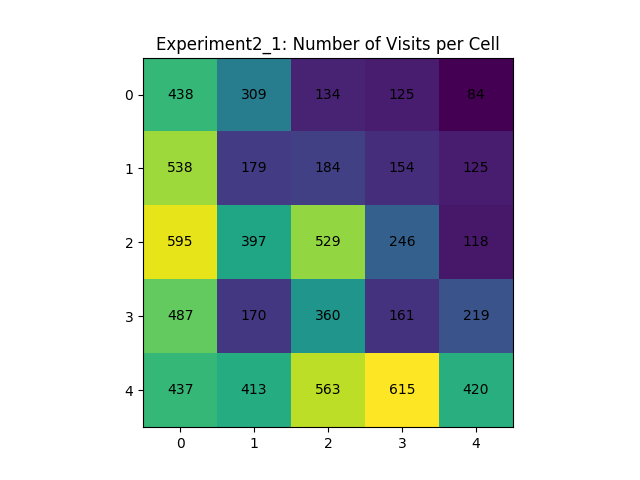
*While Agent is holding a block*



*While Agent is not holding a block*

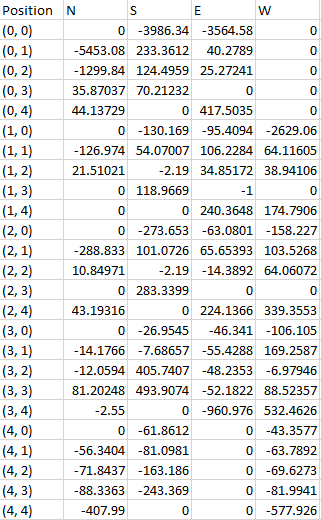
The implementation of the PEXPLOIT policy over the PGREEDY policy seemed to have improved the results for the Q Tables as problems identified in the first experiments, with regards to cells surrounding pickup and drop off locations not having the expected Q values seems to have been resolved. For example, cell (3,0)’s action for south is the highest valued action of the available actions as is expected.

This heatmap is far more indicative of the desired agent movement behavior than was found in both iterations of experiment one, proving that PEXPLOIT may prove more useful in training an agent to learn an unknown environment than PGREEDY.

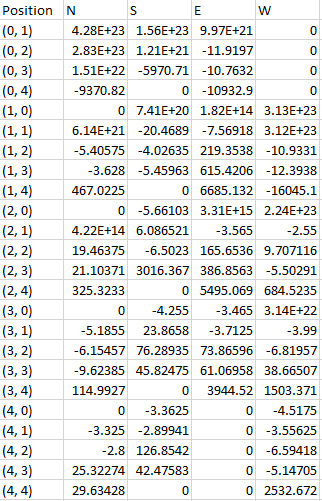


This is the first occurrence of the heatmaps corresponding to the larger state space and the smaller state space resembling one another. Heavy trafficked areas in both of the above heatmaps are as expected (as they are located between pickup and drop off locations)

**Experiment 2: Second Run**

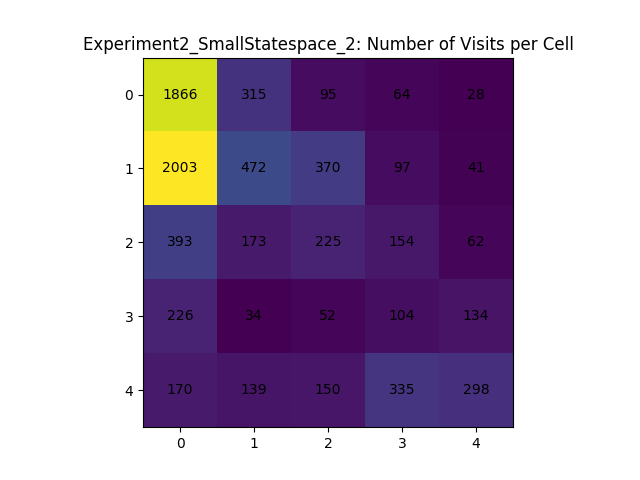


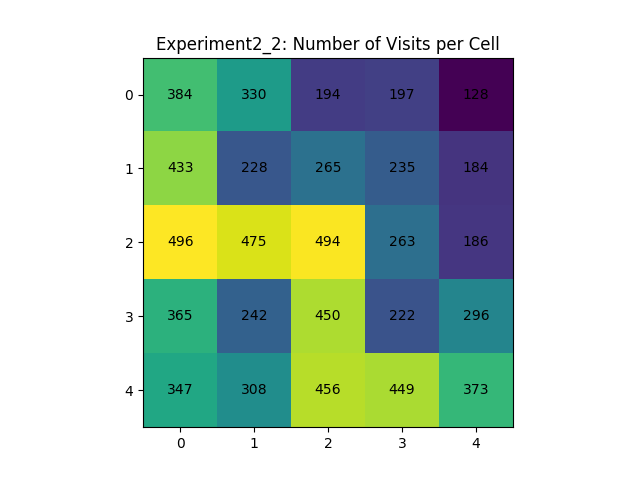
*While Agent is holding a block*



*While Agent is not holding a block*

The above Q Tables appear similar to the Q Tables obtained from the first iteration of this experiment however there are key differences. For example, the cells surrounding (0,0), a pickup location have positive Q values whereas in the first iteration of the experiment they were largely negative, however the desired behavior, the desired action leading towards a pickup location has the highest values so that much is at least clear.



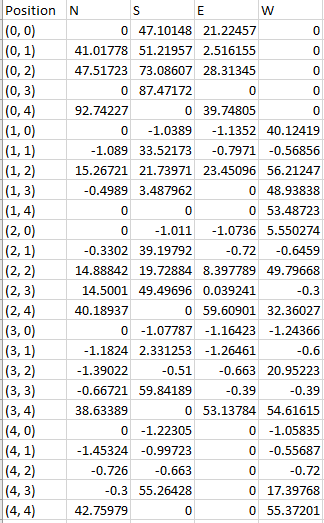
Any similarities previously mentioned end here though, as the problems thought to have been solved with the introduction and implementation of the PEXPLOIT policy from the PGREEDY policy seem to have resurfaced. The agent once again seems to have gotten stuck in a perpetual cycle alternating between two states. Perhaps the policy change was not attributable to the first iteration’s desirable output.

This heatmap is once again indicative of desired agent movement behavior and is similar to previous heatmaps associated with the wider state space, albeit with slightly different values.

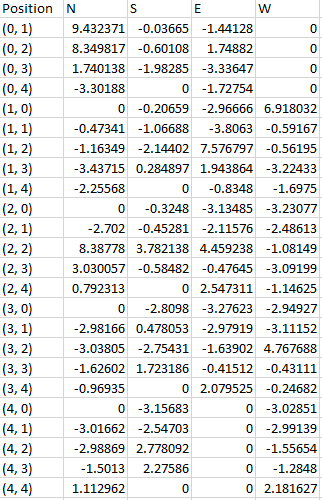
**Experiment 3**

For the third experiment our group used an αvalue of 0.3 and a γvalue of 0.5. Using these values, we applied our implementation of the SARSA Q-Learning algorithm for 200 steps using the PRANDOM policy, followed by 7800 steps using the PEXPLOIT policy.

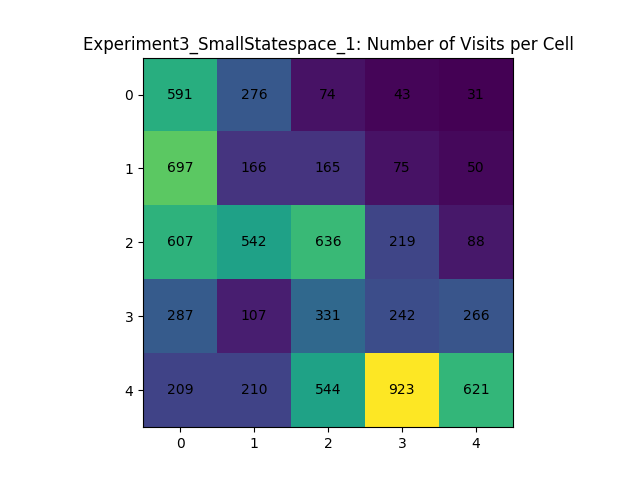
**Experiment 3: First Run**



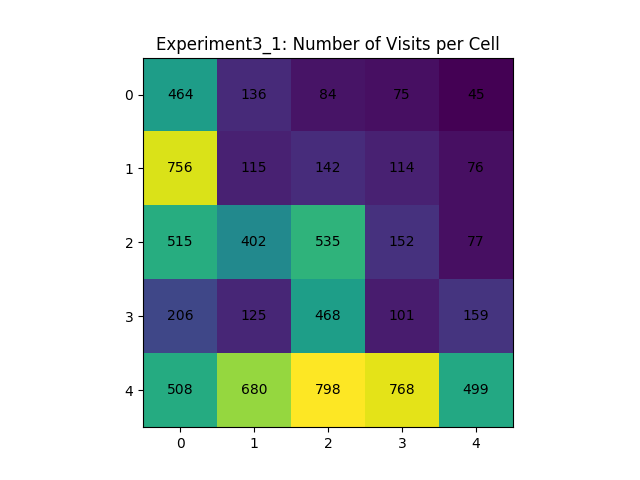
*While Agent is holding a block*



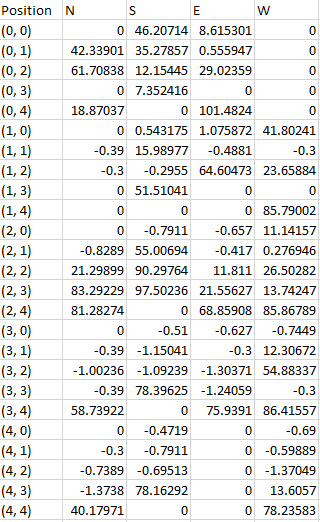
*While Agent is not holding a block*

It appears as though the some of the issues largely plaguing experiment one and but also experiment two are resolved, possibly in part to the switch to the utilization of the SARSA algorithm. For example, in the left Q Table, whilst in cell (3,0) the agent should be incentivized to move south to the drop off location in cell (4,0) while holding a block. In the previous two experiments this was not the case, as the other actions (East and West) offered higher rewards. However, in this experiment, the expected action while in this particular scenario, the action associated with moving South having the highest final Q value is present. This can be attributed to the differences inherent in the two different learning algorithms, SARSA and Q-Learning, specifically SARSA chooses an action with regards to the current policy whereas Q-Learning exhibits greedy characteristics by always choosing the action that yields the highest Q value. Examples of the aforementioned scenario are present throughout all of the previously mentioned problem areas experienced in the first two experiments.

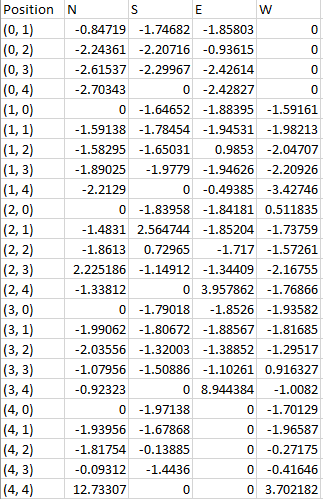
The efficacy of the SARSA algorithm over the Q-Learning algorithm is also evident here, in the analysis of the heatmap. In the previous experiments, the heat maps exhibited agent behavior that was less than ideal. The agent seemed to have gotten stuck in a perpetual loop, iterating between two states, whereas in this experiment the heat map indicates the agent’s movement was much more evenly distributed and aligns with our expectations of the agent’s movement pattern. As expected, parts of the world which do not have pickup or drop off locations are much less frequently visited, as is apparent in cell (0,4) above, whereas cells that have pickup and drop off locations are more frequently visited as can be seen in cells (4,0), (2,2), and (0,0), for example. However, it seems the agent still prefers some locations over others as certain targeted locations seem to have been forgotten, like cell (1,4).

Again, this heatmap, representative of the wider state space is indicative of the desired agent behavior, albeit with differing results and frequencies than exhibited by previous runs and algorithms. However, like the previous depicted heatmap, certain states seemed to have been overlooked.

**Experiment 3: Second Run**

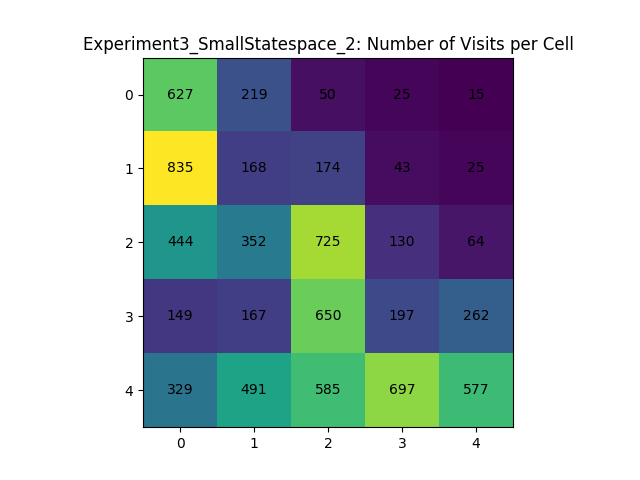


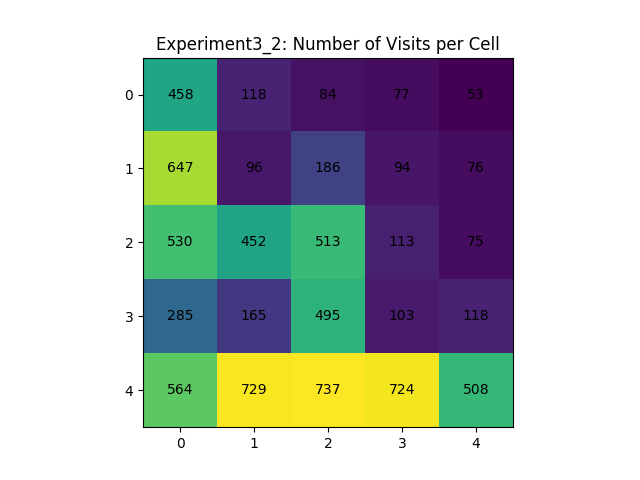
*While Agent is holding a block*



*While Agent is not holding a block*

By glancing at the above Q Tables, the results seem to be similar to the first execution of this experiment, however if you view the table on the right, which corresponds to the Q values when the agent is not holding a block, there are some glaring differences. The Q values for actions in cells near the pickup location at (0,0) offer positive rewards, while the same Q Table in the first iteration of this experiment had large, negative values. However, both of these share similarities in that the expected action to be taken by the agent, West towards the pickup location at cell (0,0) (from cell (0,1)) is untouched. The 0 value here indicates the action as never updated from its initialization at 0. This is intriguing as this should offer up the greatest reward, but as evidenced above this is not the case.



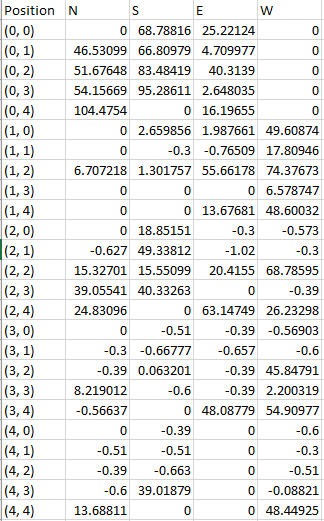
The heatmap is not as representative of desired behavior by the agent, as it slightly resembles the problem areas experienced by the first two experiments. However, it is far superior the results of experiment 1.

This heatmap corresponding to the bigger state space, is also highly indicative of ideal agent movement and behavior. It is similar to previous experiments’ results.

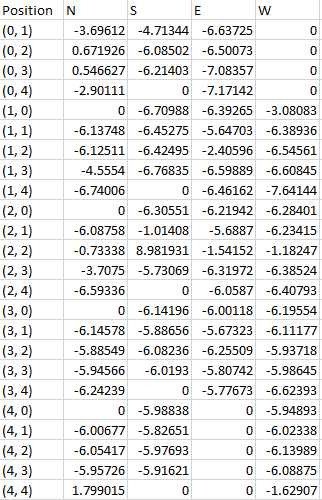
**Experiment 4**

For the fourth experiment our group used an αvalue of 0.3 and a γvalue of 1.0. Using these values, we applied our implementation of the SARSA Q-Learning algorithm for 200 steps using the PRANDOM policy, followed by 7800 steps using the PEXPLOIT policy.

**Experiment 4: First Run**

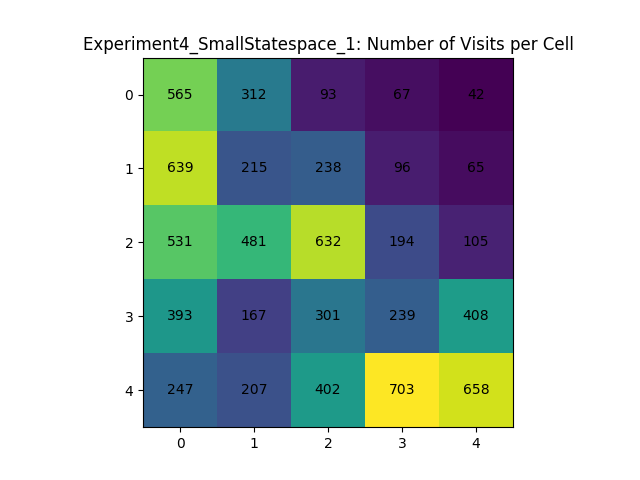


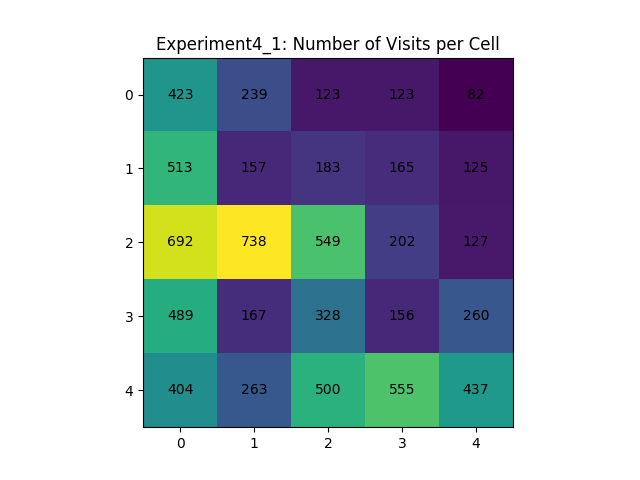
*While Agent is holding a block*



*While Agent is not holding a block*

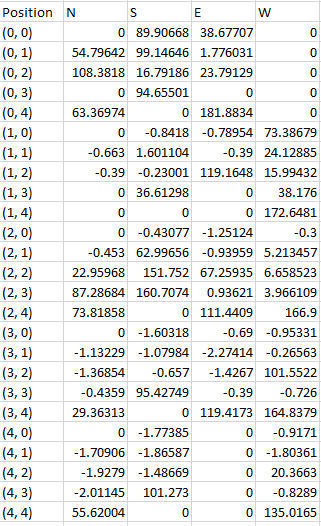
The main aspect we would like to focus on between the previous experiment, experiment 3 and this experiment is the role the discount rate has on the agent as he is learning the world. The results seem to be consistent with experiment three, as the Q values in both Q Tables seem to correspond to the expected outcomes. Although, here the Q values in both Q Tables seem to closer in values, or less of a range of values, than the previous experiment.



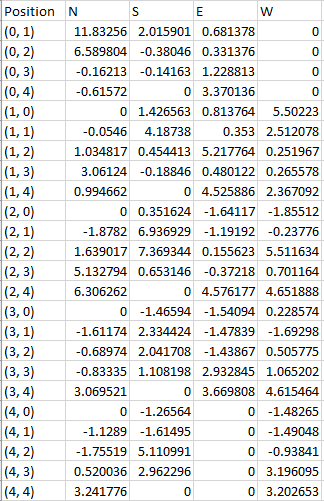
This heatmap is largely similar to its corresponding heatmap in experiment three, as the top right part of the map has been visited much less frequently than the rest, although it seems during this experiment the agent more heavily visited the cells in the bottom right part of the map, travelling between the pickup location in (4,4) and the drop off location in (4,2).

This heatmap, like the others from the larger state space is indicative of desired agent behavior, with a wide distribution of visits per cell.

**Experiment 4: Second Run**

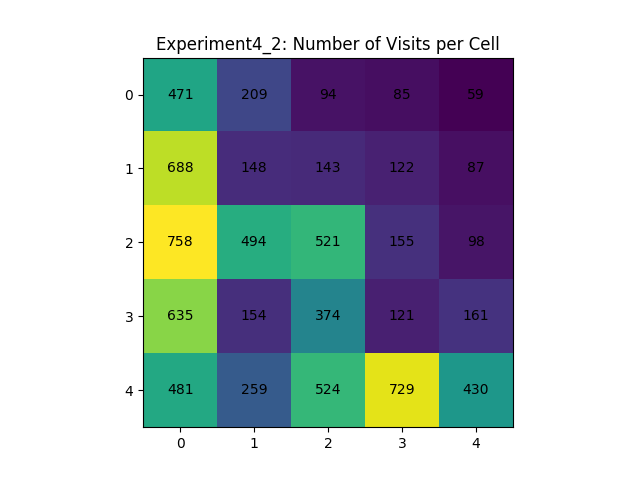


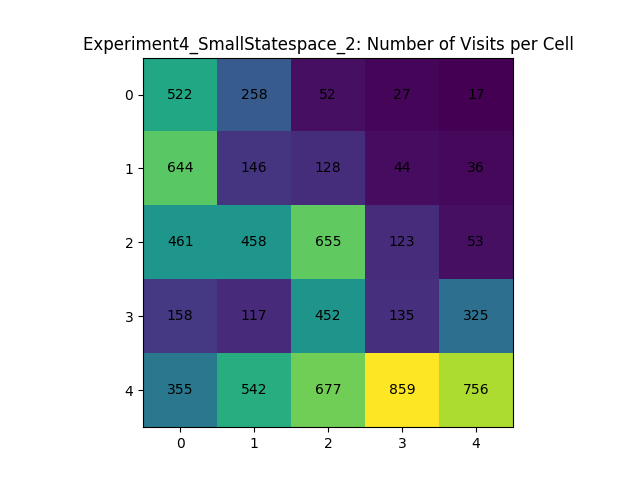
*While Agent is holding a block*



*While Agent is not holding a block*

The results from the second iteration of experiment four are similar to that of the first iteration, however in the Q Table on the right it is apparent that the actions associated with cell (4,4) are much closer in value than those in the first iteration. There are several of these inconsistencies between the two iterations of the experiments. As such, the results of the experiment, in our opinion are more desirable due to the range in Q values between states.

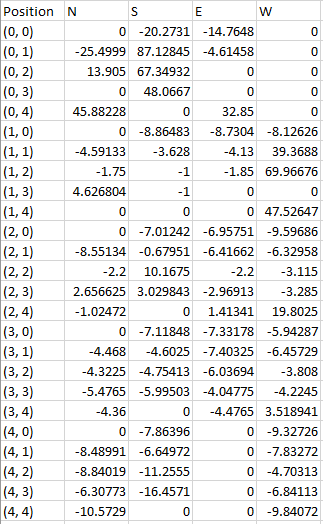


Similar to the first iteration of this experiment the agent’s movement behavior matches expectations.

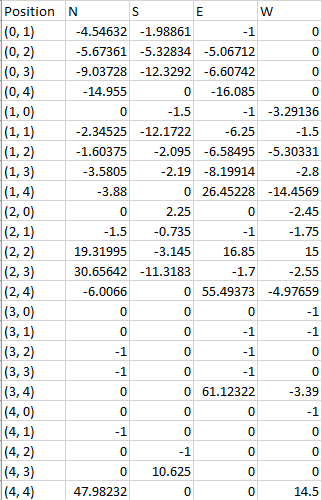
**Experiment 5**

For the fifth experiment our group used an αvalue of 0.3 and a γvalue of 0.5. Using these values, we applied our implementation of the Q-Learning algorithm for 200 steps using the PRANDOM policy, followed by 7800 steps using the PEXPLOIT policy, similar to experiment 2. However, in this experiment after the agent reached the terminal state twice, the pickup and drop off locations were swapped.

**Experiment 5: First Run**

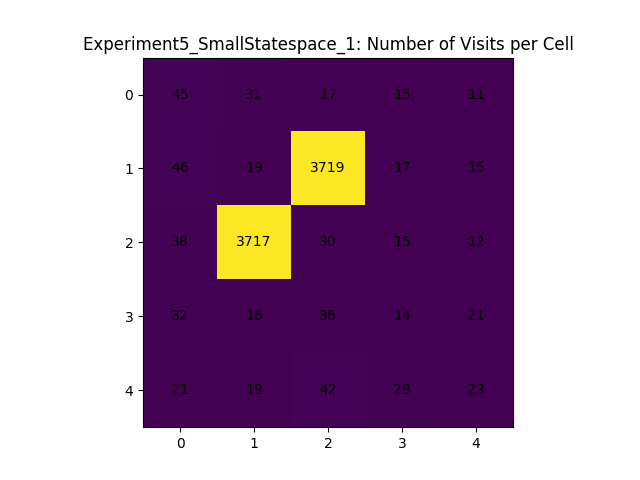
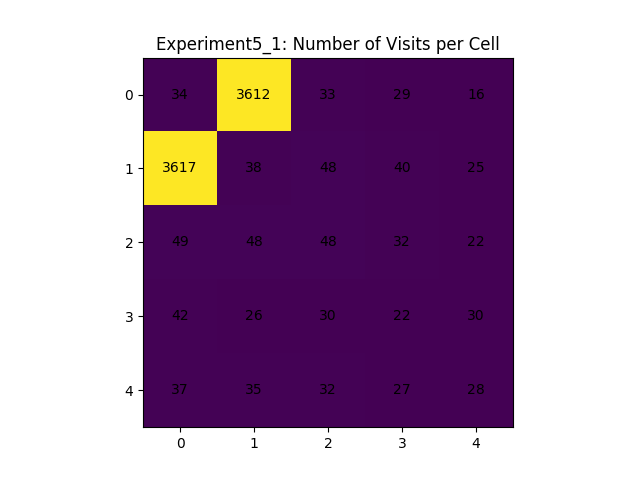


*While Agent is holding a block*



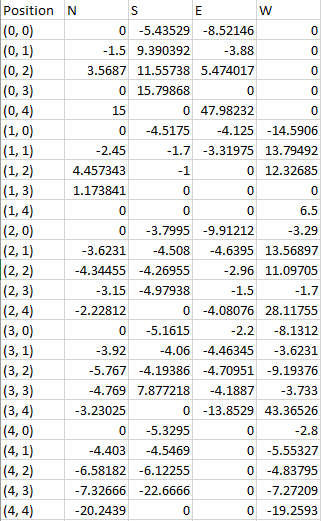
*While Agent is not holding a block*

During this experiment, as mentioned above, during the course of the experiment the pickup and drop off locations were switched. This caused our agent to exhibit highly undesirable behavior as there are several cells (locations) that seem to have gone untouched (as evidenced by Q values of 0 in the Q Tables above)

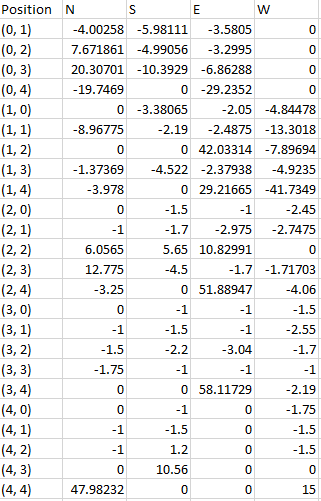


Both of theses heatmaps are indicative of the undesirable movement behavior exhibited by our agent during this experiment. It seems that the swapping of the pickup and drop off locations caused our agent to get stuck in loops, some how circumnavigating the cardinal directional movement pattern, which indicates that our algorithm was perhaps incorrectly implemented.

**Experiment 5: Second Run**

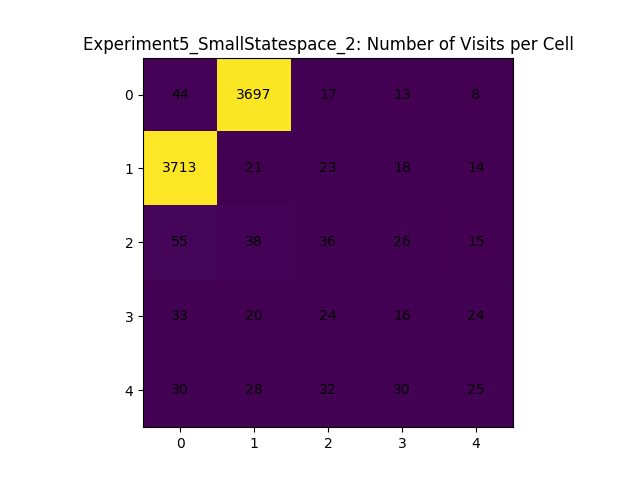
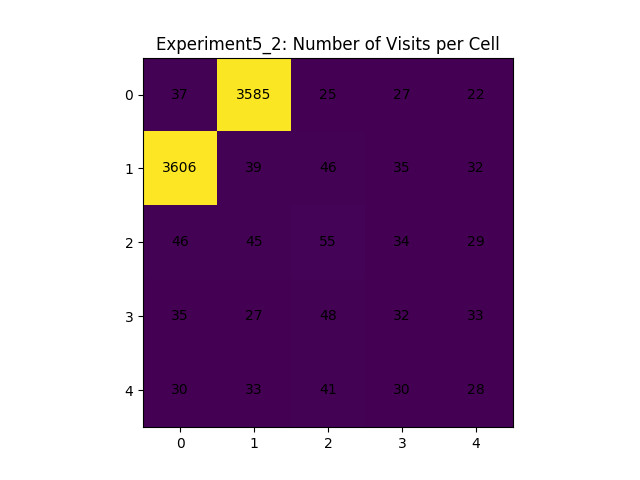


*While Agent is holding a block*



*While Agent is not holding a block*

The second iteration of this experiment exhibits better, albeit still undesirable results, as there are less untouched states but there are still numerous examples. Again, these results seem to indicate a problem in our implementation as our agent was unable to adapt to the swapping of the pickup and drop off locations.

Both of the above heatmaps display nearly identical results to those found in the previous iteration of experiment five, including the undesirable movement behavior of our agent.

**Conclusion**

As is evidenced by the analysis above, experiments three and four produced the best results. Both of these experiments implemented the SARSA Q-Learning algorithm, which in our opinion based on the above results, indicated the most desirable agent movement behavior and Q values which signified our expectations of actions leading to the targeted pickup and drop off location, depending on the state of the agent (whether or not it was holding a block) were met. However, experiment two, which consisted of the use of the Q-Learning algorithm alongside the PEXPLOIT policy also produced desirable results, akin to those produced by experiments three and four. The agent, in some cases, was not functioning as fully intending and we are unsure whether it is attributable to an incorrect implementation or simply the combination of the learning algorithm used alongside the policies and variables used, however this project has served as an exciting, albeit brief introduction the to reinforcement learning and has truly been eye opening.