MIE567 Assignment 2

Model-Free RL

Jade Khiev 1000751616
Nikita Kochnev 1001329378
Padmanie Maulkhan 996769616

Flags Domain

Part A - Modelling

				G
	2			
3				1
S				4

Question 1a) The states have been defined as follows: S(f, x, y) where (f) is how many flags the robot currently has ranging from 0-4, (x) is the xth row with 1 being defined as the leftmost column and (y) is the yth column with 1 being defined as the top row. Therefore, the top left space with 0 flags would be defined as S(0, 1, 1). The full state space is shown in the diagram below.

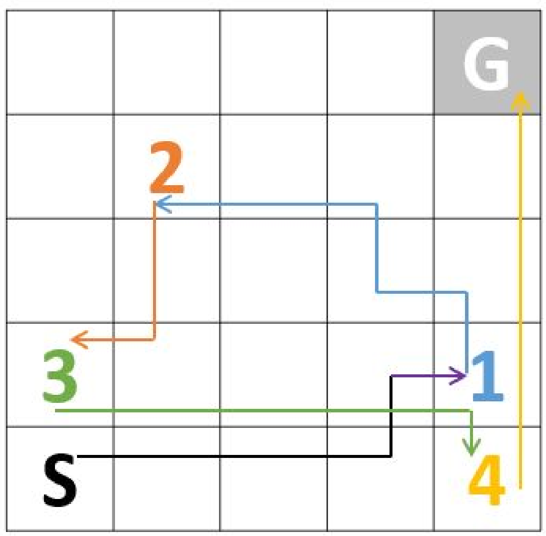
```
states = [
        (0,1,1),    (1,1,1),    (2,1,1),    (3,1,1),    (4,1,1),
        (0,1,2),    (1,1,2),    (2,1,2),    (3,1,2),    (4,1,2),
        (0,1,3),    (1,1,3),    (2,1,3),    (3,1,3),    (4,1,3),
        (0,1,4),    (1,1,4),    (2,1,4),    (3,1,4),    (4,1,4),
        (0,1,5),    (1,1,5),    (2,1,5),    (3,1,5),
        (0,2,1),    (1,2,1),    (2,2,1),    (3,2,1),    (4,2,1),
        (0,2,2),                (2,2,2),    (3,2,2),    (4,2,2),
        (0,2,3),    (1,2,3),    (2,2,3),    (3,2,3),    (4,2,3),
        (0,2,4),    (1,2,4),    (2,2,4),    (3,2,4),    (4,2,4),
        (0,2,5),    (1,2,5),    (2,2,5),    (3,2,5),    (4,2,5),
        (0,3,1),    (1,3,1),    (2,3,1),    (3,3,1),    (4,3,1),
        (0,3,2),    (1,3,2),    (2,3,2),    (3,3,2),    (4,3,2),
        (0,3,3),    (1,3,3),    (2,3,3),    (3,3,3),    (4,3,3),
        (0,3,4),    (1,3,4),    (2,3,4),    (3,3,4),    (4,3,4),
        (0,3,5),    (1,3,5),    (2,3,5),    (3,3,5),    (4,3,5),
        (0,4,1),    (1,4,1),                (3,4,1),    (4,4,1),
        (0,4,2),    (1,4,2),    (2,4,2),    (3,4,2),    (4,4,2),
        (0,4,3),    (1,4,3),    (2,4,3),    (3,4,3),    (4,4,3),
        (0,4,4),    (1,4,4),    (2,4,4),    (3,4,4),    (4,4,4),
                    (1,4,5),    (2,4,5),    (3,4,5),    (4,4,5),
        (0,5,1),    (1,5,1),    (2,5,1),    (3,5,1),    (4,5,1),
        (0,5,2),    (1,5,2),    (2,5,2),    (3,5,2),    (4,5,2),
        (0,5,3),    (1,5,3),    (2,5,3),    (3,5,3),    (4,5,3),
        (0,5,4),    (1,5,4),    (2,5,4),    (3,5,4),    (4,5,4),
        (0,5,5),    (1,5,5),    (2,5,5),                (4,5,5),

        (5,1,5) # TERMINAL STATE
        ]
```

The actions have been defined as follows: A(a) where a is one of four possible actions which are: Up, Down, Left and Right. If an action would take you outside the boundary or if you are in the terminal state, you remain your current state.

Question 1b) The reward function was constructed with the following assumptions in mind: Firstly, there would be punishment for visiting any of the flag states prematurely. Secondly, there would be no punishment revisiting a state that previously held a captured flag. These exist because once a flag has been collected, it is an empty square and should be treated as such and in order to ensure the robot collects the flags in order we need to penalize incorrect ordering. The reward values are as follows:

- Reward of -10 for visiting a flag state in the wrong order (2/3/4/G before 1, 3/4/G before 2 and so on)
- Reward of -1 for any move that does not land in flag state. We impose penalty in order to incentivise completing the goal as quickly as possible. This includes moves that hit the boundary.
- Reward of +25 for visiting a flag state in the correct order (1 then 2, 2 then 3 and so on).
    - The large reward incentivizes visiting these states as quickly as possible in the correct order.

Question 1c) Based on the Flags domain, a potential optimal policy is drawn below:

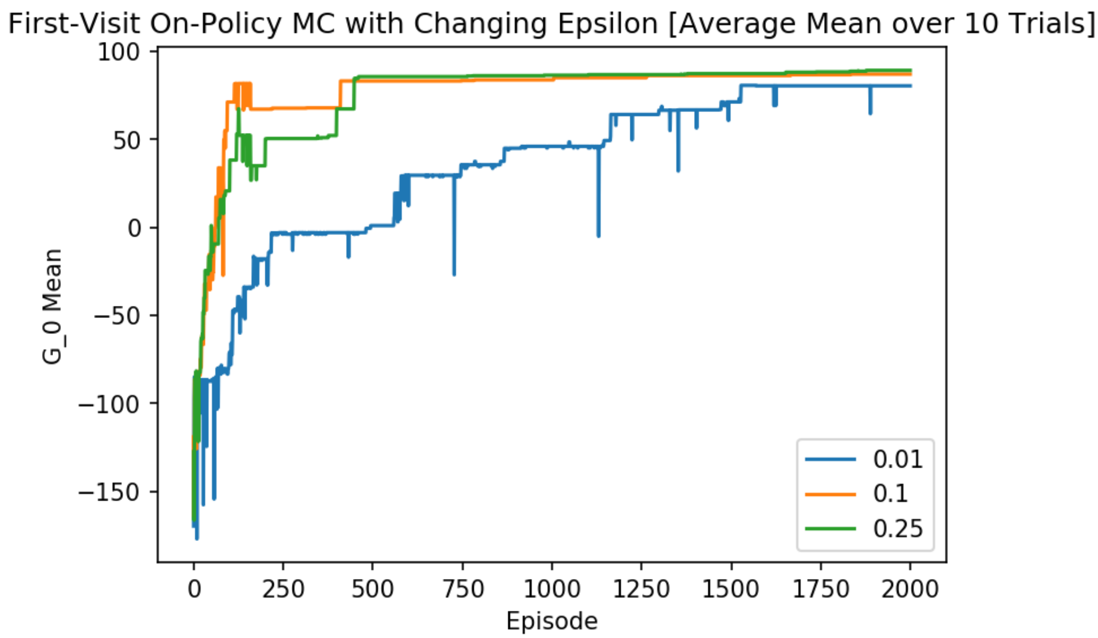


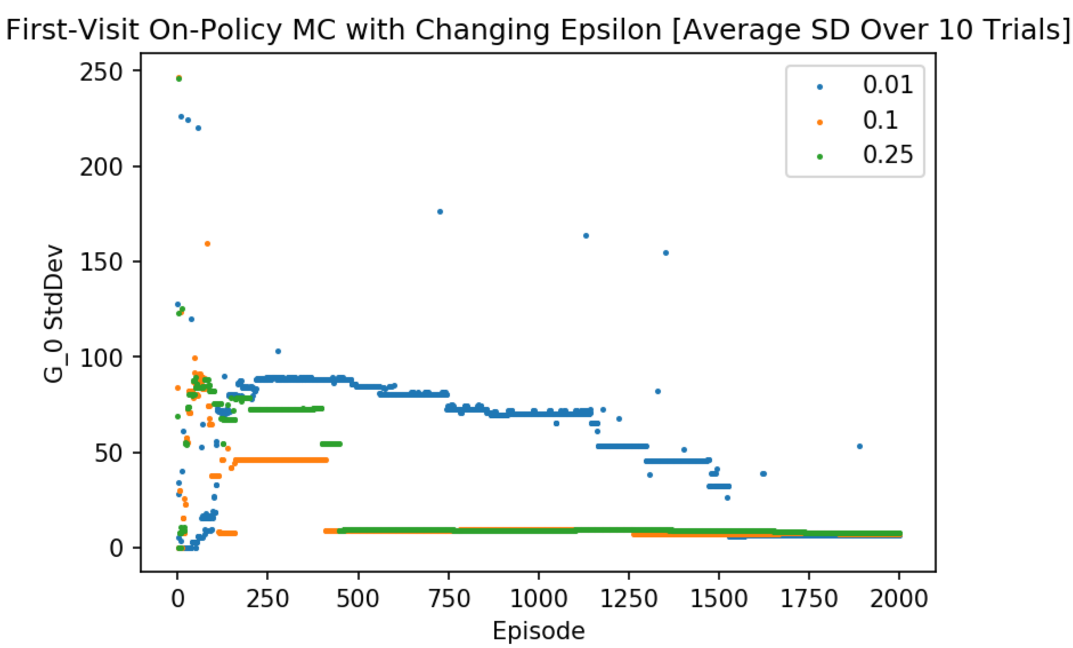
There is more than one optimal policy because there are several points of indifference in which different actions result in the same travel length. For example, going from Flag 1 to Flag 2 the minimum amount of moves is 5 but there are various 5 move paths to get you to Flag 2. The diagram above represents one of the optimal policies.

Question 2) Please refer to Flags.py for the implementation of the flags environment.

Part B - On-Policy First-Visit Monte Carlo Control

10 trials were run for this method and the sample mean and standard deviation were plotted for varing values of epsilon.



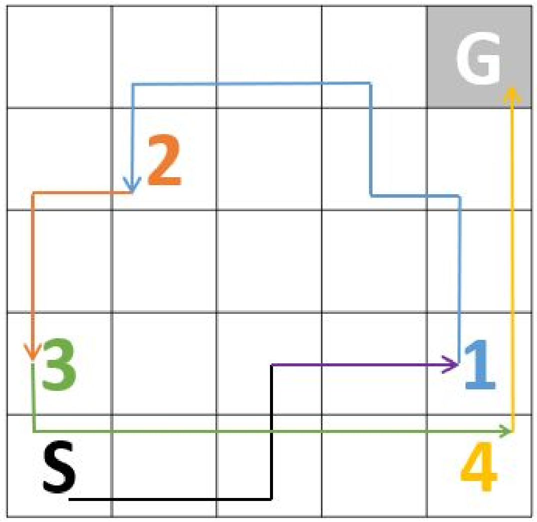


Average Q-Values Over 10 Trials (Epsilon = 0.25)

The policy based on the average of 10 trials gets stuck between steps 17 and 18. The yellow highlighted box in state (3,5,3) represents the second highest value, which is likely to sometimes evaluate to most desirable option. To keep the policy moving along, step 19 onwards shows what the policy would be when the most optimal option is to make a right from state (3,5,3). Alternatively, state (3,4,3)'s second highest action's value could have also been selected to show policy. The full Q[S][A] table for MC can be found in Appendix I.

State\Action	U	D	L	R	Step
(0, 5, 1)	17.4248	6.3076	19.7072	63.1612	1
(0, 5, 2)	42.2204	38.9289	26.3825	59.54	2
(0, 5, 3)	66.1168	56.7154	21.0933	61.9313	3
(0, 4, 3)	11.5032	50.4754	49.1856	75.3391	4
(0, 4, 4)	67.9079	68.7719	50.3728	82.5177	5
(1, 4, 5)	28.9049	33.0797	41.2143	28.4055	6
(1, 4, 4)	47.4052	40.2019	48.3859	0.4727	7
(1, 4, 3)	58.5572	14.8042	44.7792	39.572	8
(1, 3, 3)	69.1665	48.867	62.452	59.1322	9
(1, 2, 3)	52.9597	51.4027	74.3055	40.9163	10
(2, 2, 2)	40.1659	42.6975	49.6504	41.7735	11
(2, 2, 1)	37.6123	52.6743	34.251	40.4942	12
(2, 3, 1)	45.264	56.4607	42.3346	44.8199	13
(3, 4, 1)	23.6856	30.3948	22.8413	25.8424	14
(3, 5, 1)	26.9213	25.6579	22.91	30.4884	15
(3, 5, 2)	28.032	14.6375	24.7798	32.4071	16
(3, 5, 3)	35.3389	18.1458	21.3222	34.4483	17
(3, 4, 3)	25.2624	34.0668	25.8725	33.4889	18
(3, 5, 4)	36.9929	39.5472	35.5435	43.9885	19
(4, 5, 5)	19.2676	16.6477	15.7184	15.1637	20
(4, 4, 5)	21.1362	16.7103	17.1675	17.7427	21
(4, 3, 5)	22.908	19.4085	19.4479	17.8985	22
(4, 2, 5)	25	21.1212	21.2685	23.0995	23
(5, 1, 5)	0	0	0	0	24

Final Policy for MC

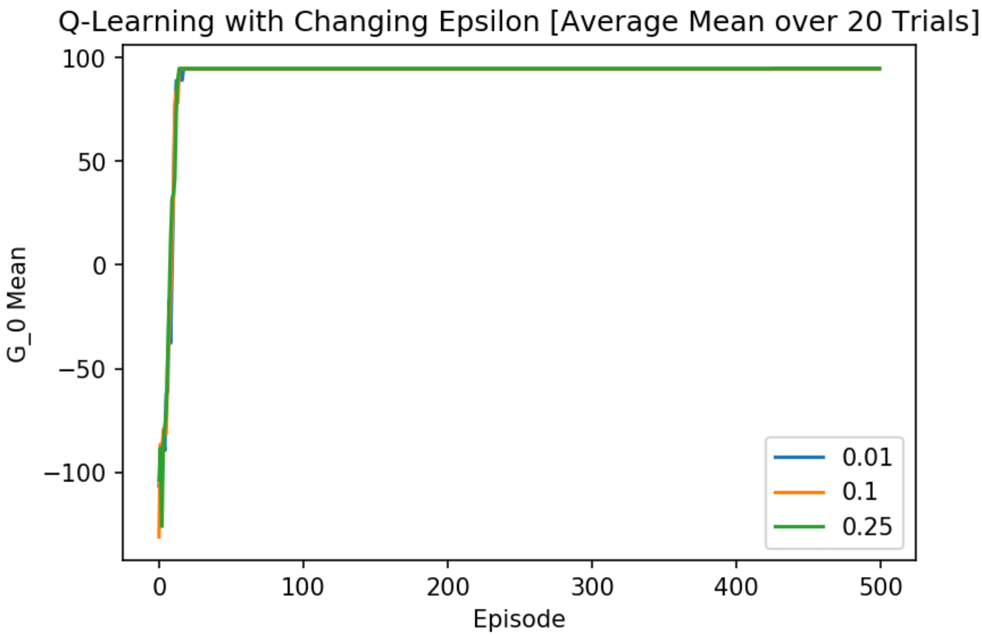


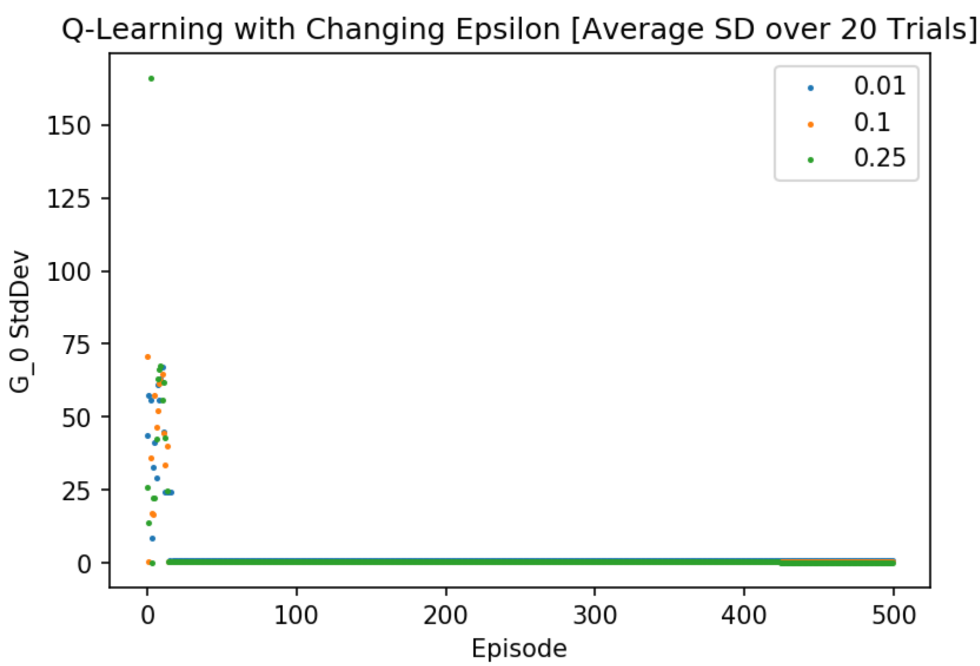
Questions:

1. *Best value of Epsilon and why:* Looking at the sample mean and variance graphs it appears that the best value of epsilon is slightly 0.25 because it yields the highest mean and lowest variance.
2. *Convergence:* There is convergence for each value of epsilon as demonstrated by the sample mean plot. At around 1500 episodes, all the epsilon values stabilize.
3. *Final Policy and Q-Values:* The final policy and Q-values are shown above.
4. *Does the correspond to an optimal policy:* This is **not an optimal policy** due to the fact taken from Flag 1 to Flag 2 is suboptimal. However, it is worth noting that the other paths (S-1, 2-3, 3-4, 4-5) do follow optimal routes.

Part C - Q-Learning (Off-Policy TD Learning)

20 trials were run for this method and the sample mean and standard deviation were plotted for varying values of epsilon.



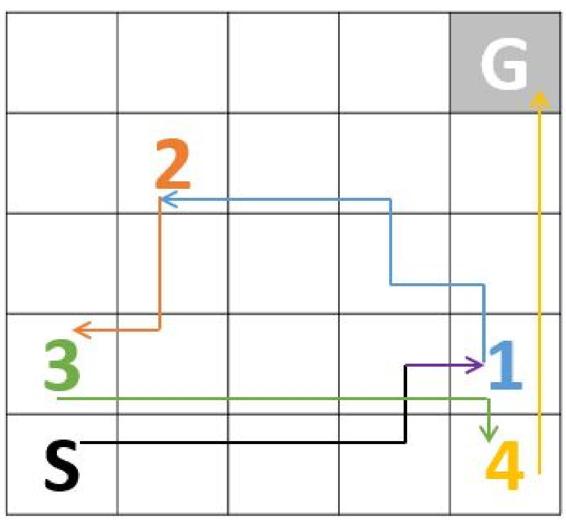


Average Q Values Over 20 Trials (Epsilon = 0.01)

The full Q[S][A] table for Q-Learning can be found in Appendix II.

State\Action	U	D	L	R	Step
(0, 5, 1)	-3.6921	30.3965	24.8763	94.8739	1
(0, 5, 2)	77.7747	20.8123	13.3312	26.5210	2
(0, 4, 2)	2.5009	18.1903	-0.1432	77.5683	3
(0, 4, 3)	11.3184	9.3015	21.5023	97.2691	4
(0, 4, 4)	4.6335	6.1713	7.2323	102.8677	5
(1, 4, 5)	10.7015	-4.4044	67.0422	16.5114	6
(1, 4, 4)	8.3504	-1.1786	63.9705	23.1412	7
(1, 4, 3)	47.7164	8.7544	20.6007	0.7421	8
(1, 3, 3)	9.2999	12.7006	71.5267	2.3193	9
(1, 3, 2)	82.8092	16.0972	9.4084	18.6464	10
(2, 2, 2)	11.2517	56.7283	9.0141	-0.8210	11
(2, 3, 2)	23.1831	9.2434	57.2541	1.7491	12
(2, 3, 1)	3.8922	64.1350	18.0009	15.4803	13
(3, 4, 1)	-1.6526	11.6516	12.4170	36.7180	14
(3, 4, 2)	-1.0951	28.7003	18.2970	17.0809	15
(3, 5, 2)	0.0324	2.9990	2.3445	39.8044	16
(3, 5, 3)	9.3318	11.3797	14.5859	43.7206	17
(3, 5, 4)	5.9364	25.7214	23.3057	46.0371	18
(4, 5, 5)	21.2495	9.9359	-0.5973	6.9551	19
(4, 4, 5)	22.4736	7.3331	-0.7258	12.0574	20
(4, 3, 5)	23.7109	7.3096	1.8546	2.9716	21
(4, 2, 5)	24.9061	2.6757	0.8256	9.8385	22
(5, 1, 5)	0.0000	0.0000	0.0000	0.0000	23

Final Policy for Q Learning

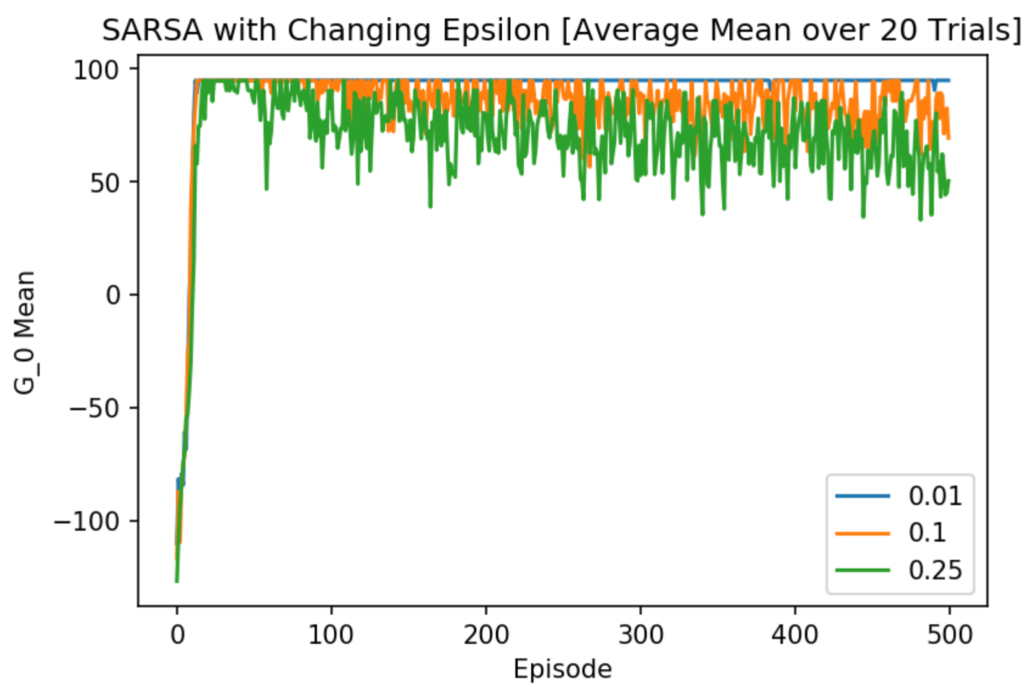


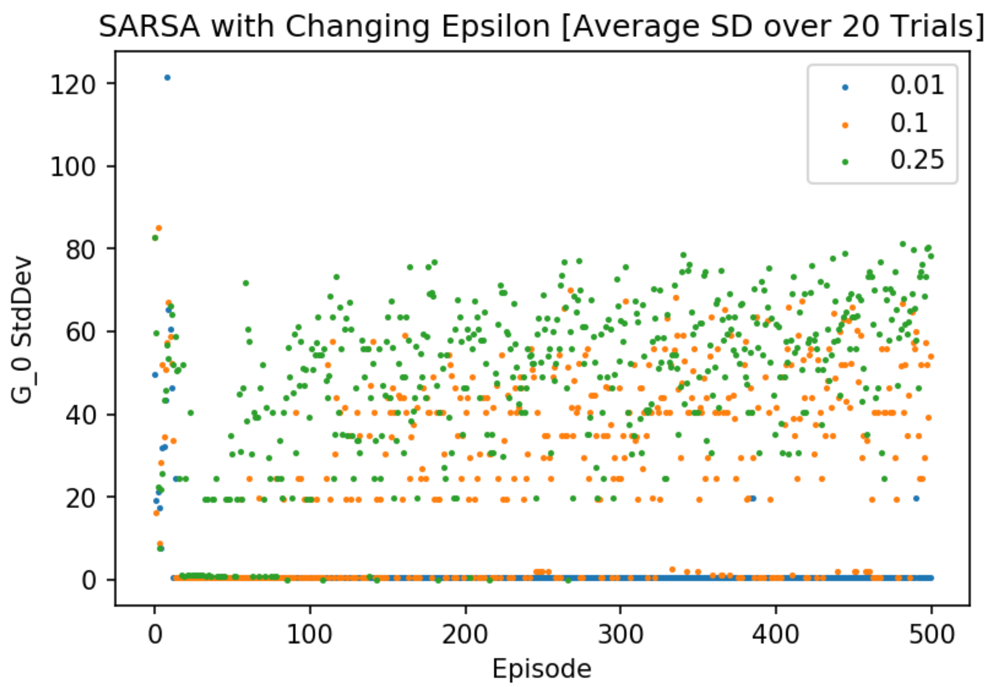
Questions:

1. *Best value of Epsilon and why:* Looking through the sample means and variances, there is almost no significant difference between the various epsilon values. Therefore, all 3 values are equally strong.
2. *Convergence:* For all 3 values of epsilon there was convergence as is evident by the sample mean graph. All the 3 values settled quickly and at the same value.
3. *Final Policy and Q-Values:* The final policy and Q-values are shown above.
4. *Does this correspond to an optimal policy:* This policy does correspond to one possible optimal policy.

Part D - SARSA (On-Policy TD Learning)

20 trials were run for this method and the sample mean and standard deviation were plotted for varying values of epsilon.



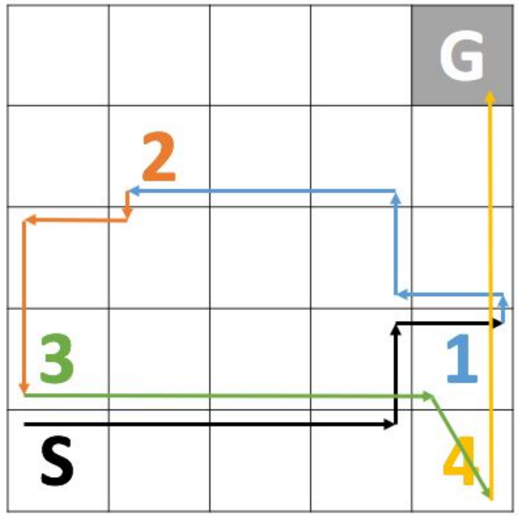


Average Q-Values Over 20 Trials (Epsilon = 0.01)

The full Q[S][A] table for Q-Learning can be found in Appendix III.

State\Action	U	D	L	R	Step
(0, 5, 1)	-4.333	27.594	31.811	89.664	1
(0, 5, 2)	17.456	22.019	22.422	74.768	2
(0, 5, 3)	12.251	34.792	0.898	73.446	3
(0, 5, 4)	85.33	26.017	22.996	2.168	4
(0, 4, 4)	0.209	27.026	4.968	98.935	5
(1, 4, 5)	5.975	-4.607	72.024	24.941	6
(1, 4, 4)	24.889	-0.796	59.131	11.15	7
(1, 4, 3)	61.621	2.772	4.333	3.608	8
(1, 3, 3)	50.117	7.737	18.276	9.811	9
(1, 2, 3)	-0.143	19.252	72.236	5.408	10
(2, 2, 2)	-1.1	12.879	53.58	3.343	11
(2, 2, 1)	1.135	58.391	6.216	8.598	12
(2, 3, 1)	24.298	63.204	28.539	6.569	13
(3, 4, 1)	-1.651	2.271	2.242	39.508	14
(3, 4, 2)	1.632	29.401	4.735	12.934	15
(3, 5, 2)	13.075	1.745	3.207	37.024	16
(3, 5, 3)	2.594	4.831	6.432	43.452	17
(3, 5, 4)	0.592	16.272	8.604	45.935	18
(4, 5, 5)	21.227	1.432	-1.397	8.278	19
(4, 4, 5)	22.495	-0.843	-0.666	11.634	20
(4, 3, 5)	23.747	4.227	0.314	8.818	21
(4, 2, 5)	25	7.967	0.948	1.824	22
(5, 1, 5)	0	0	0	0	23

Final Policy for SARSA

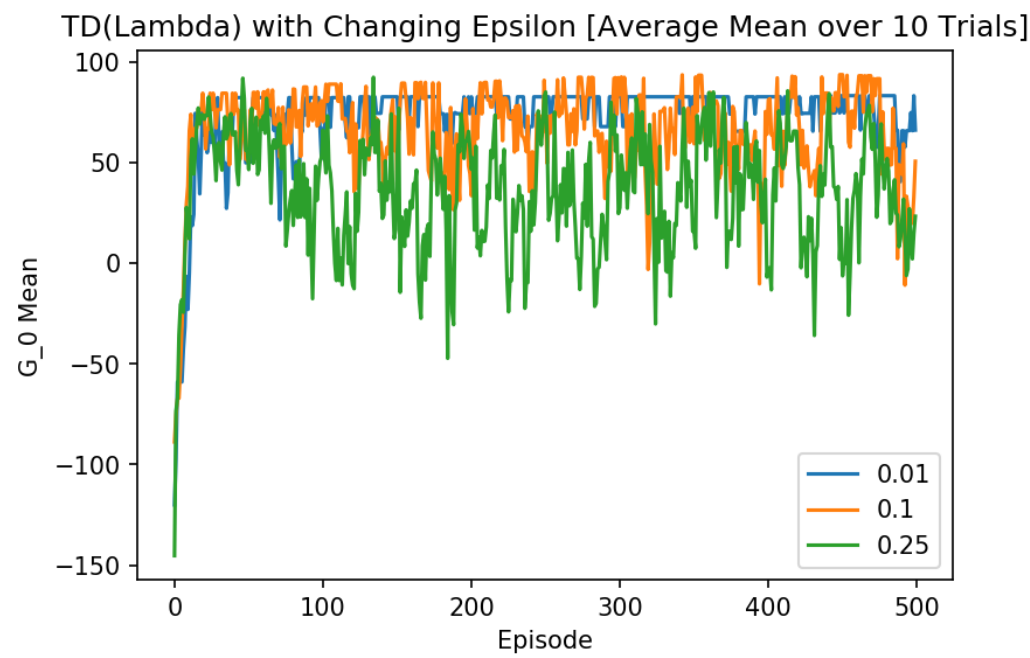


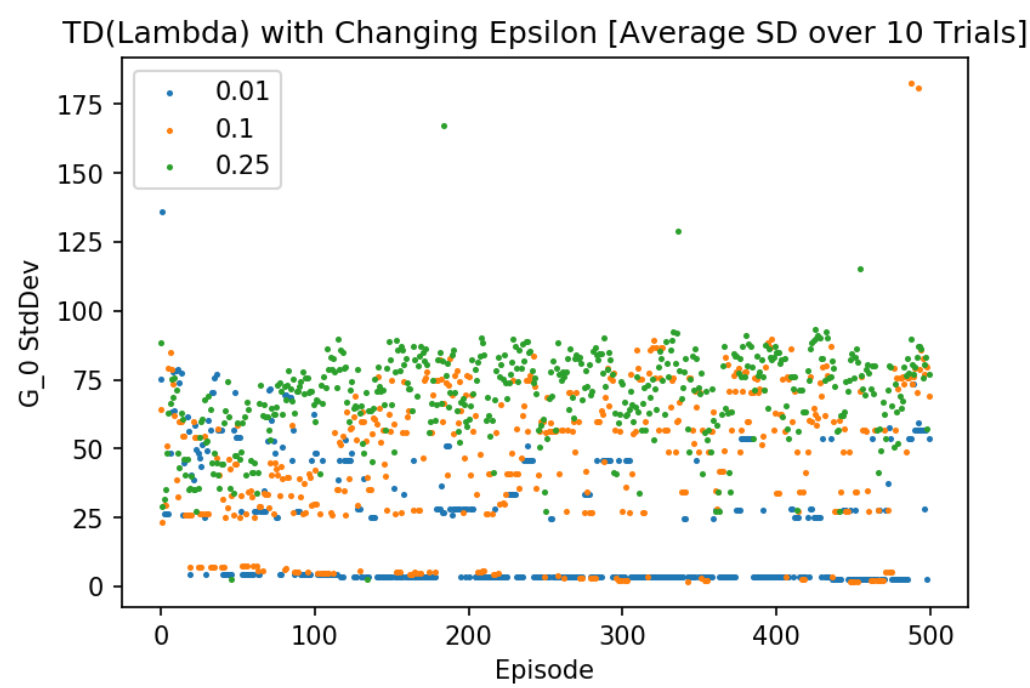
Questions:

1. *Best value of Epsilon and why:* Based on the mean and variation graphs, we would have to say that the best value is 0.01 since it corresponds to the highest sample mean and lowest variance consistently over the episodes.
2. *For each value of Epsilon was their convergence:* Convergence is not very clear for all values of epsilon since they do not settle on any particular G_0 value. Epsilon of 0.01 seems to converge quickly but as the episodes reach 300+ it dips down. It possible they are approaching convergence towards the end but it is unclear.
3. *Final Policy and Q-Values:* The final policy and Q-values are shown above.
4. *Does this correspond to an optimal policy:* This policy does correspond to one possible optimal policy.

Part E - TD(λ)

10 trials were run for this method and the sample mean and standard deviation were plotted for varying values of epsilon.



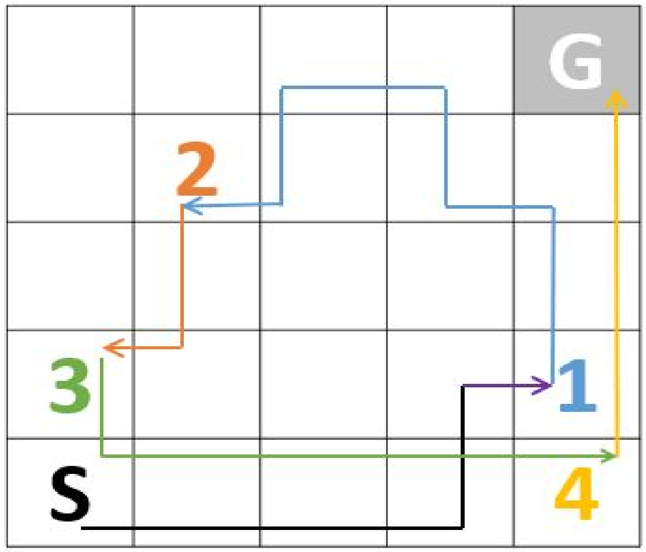


Average Q-Values Over 10 Trials (Epsilon = 0.01)

The policy based on the average over 10 trials gets stuck between steps 9 and 10. The highlighted yellow box in state (1,4,3) indicates the second highest value, which is likely to sometimes evaluate to most desirable option. To keep the policy moving along, steps 11 onward shows the most optimal policy if the agent was to move left from state (1,4,3). The full Q[S][A] table for TD(lambda) can be found in Appendix IV.

State\Action	U	D	L	R	Step
(0, 5, 1)	23.562	27.698	16.516	83.222	1
(0, 5, 2)	52	1.18	13.96	52.916	2
(0, 5, 3)	11.989	15.125	-0.558	70.109	3
(0, 5, 4)	69.199	64.059	31.096	4.133	4
(0, 4, 4)	9.411	19.494	35.607	89.513	5
(1, 4, 5)	44.266	7.202	47.221	27.967	6
(1, 4, 4)	51.819	11.542	5.907	4.847	7
(1, 3, 4)	30.764	18.856	53.901	14.152	8
(1, 3, 3)	26.007	39.742	31.941	9.852	9
(1, 4, 3)	35.963	10.24	34.651	-2.75	10
(1, 4, 2)	46.797	13.236	-1.954	8.883	11
(1, 3, 2)	44.588	7.862	43.405	2.906	12
(2, 2, 2)	19.558	33.289	4.882	30.049	13
(2, 3, 2)	21.436	16.399	41.446	13.628	14
(2, 3, 1)	13.641	50.326	2.422	20.638	15
(3, 4, 1)	3.8	23.252	16.414	32.135	16
(3, 4, 2)	12.332	19.784	10.097	22.762	17
(3, 4, 3)	1.94	22.586	1.843	11.534	18
(3, 5, 3)	5.188	0.25	8.187	28.112	19
(3, 5, 4)	10.391	26.775	11.68	24.483	20
(3, 4, 4)	5.876	1.137	-1.418	14.823	21
(3, 4, 5)	6.879	25.795	2.864	14.412	22
(4, 5, 5)	14.208	2.18	1.828	-5.121	22
(4, 4, 5)	11.759	-0.022	3.099	-4.625	23
(4, 3, 5)	10.964	9.54	12.912	11.489	24
(4, 3, 4)	15.774	-0.59	2.181	13.176	25
(4, 2, 4)	12.342	-0.263	3.88	6.568	26
(4, 1, 4)	11.95	6.352	7.395	23.082	27
(5, 1, 5)	0	0	0	0	28

Final Policy for TD(Lambda)



Questions:

1. *Best value of Epsilon and why:* The best value of epsilon is 0.01 based on the mean and variance plots. We can see 0.01 produces the highest mean and is accompanied by the lowest variance.
2. *For each value of Epsilon was their convergence:* Only the value of 0.01 converges based of the sample mean plot. It does have some minor instability but for the most part it does converge and stay around the same G_0 value whereas the other epsilon values do not.
3. *Final Policy and Q-Values:* The final policy and Q-values are shown above.
4. *Does the correspond to an optimal policy:* This is not an optimal policy due to the path taken from Flag 1 to Flag 2 being suboptimal. As with Monte Carlo, it is worth noting that the other routes between flags correspond to optimal paths.

Comparison of Algorithms

Part F - Comparison of Algorithms

Comparison of Generated Policy
Referring to the policies illustrated above, we can see that two algorithms produced optimal policies and two did not. Q-Learning and SARSA produced optimal policies and MC and TD(Lambda) did not. The sub-optimal policies were briefly touched on above but MC and TD(0.9) are essentially equivalent algorithms in the way they approach the problem and as such yielded similar results. By being focused strongly on exploitation and not exploration, it is likely that at some point the algorithms decided to take the current best option instead of exploring. It is also worth noting that the algorithms are off by a very small deviation; from Flag 1 to Flag 2, the rest of the policy is optimal.

If we look at the Q-values and compare them to the characteristics of their respective algorithms we can begin to understand what they mean. Q-Learning takes a risk oriented approach in which it is always looking for the best action (in theory). SARSA is a more cautious algorithm which follows the epsilon-greedy approach. Both of these algorithms produced optimal policies but the Q-values represent a different behaviour in how they got there. MC on the other hand simply produced the best value that it found over its many iterations. This is different then all other algorithms and therefore resulted in a different (sub-optimal) policy.

Comparison of Algorithm by Epsilon
Below is an extensive comparison of the three tested epsilon values and algorithm performance for each value. Convergence speed, correct convergence and variance are discussed.

Epsilon	Convergence Speed (1 being the fastest)	Quality of Policy	Variance (1 being the lowest)
0.01	1. Q-Learning 2. TD(Lambda) 3. SARSA 4. MC	1. Q-Learning 2. SARSA 3. TD(Lambda) 4. MC	1. Q-Learning 2. TD(Lambda) 3. SARSA 4. MC
0.1	1. Q-Learning 2. TD(Lambda) 3. SARSA 4. MC	1. Q-Learning 2. SARSA 3. TD(Lambda) 4. MC	1. Q-Learning 2. TD(Lambda) 3. SARSA 4. MC
0.25	1. Q-Learning 2. TD(Lambda) 3. SARSA 4. MC	1. Q-Learning 2. SARSA 3. TD(Lambda) 4. MC	1. Q-Learning 2. SARSA 3. TD(Lambda) 4. MC

There are some interesting results that can be looked at based on the plots that we generated. Firstly, the value of epsilon did not matter for Q-Learning. The same value produced the exact same mean and variance. Another expected result is that MC had the highest variance regardless of epsilon during the first 500 episodes. This is to be expected because the MC algorithm does have the highest variance due to the fact it does not use bootstrapping. Additionally, for all values of epsilon we see that MC had the slowest convergence time which also makes sense due to the fact MC requires large amounts of data in order to exhaust all options and build a policy.

We can also note that SARSA and TD(Lambda) both performed very similarly to each other in terms of mean and especially variance. Given that both of these methods are On-Policy algorithms this is a result that is not too surprising. Conversely, the single off-policy algorithm (Q-Learning) performed great. All values of epsilon converged quickly and to the same value and all had identical variance values which were close to 0.

Another interesting point regarding policy is that since we implemented TD(0.9) which is very similar to MC since TD(1)=MC. We can see that they both returned similar policies. This is an expected result since under this value of lambda they are almost the same algorithm. Interestingly enough, they both returned sub-optimal policies. Given that they are very similar, this is an interesting observation that makes sense.

The quality of the policy was unaffected by the epsilon value.

Comparison of Epsilon

Epsilon showed some similar trends across the various algorithms.

- *Fastest Convergence:* In three out of four algorithms, the epsilon value 0.01 led to the fastest convergence, as demonstrated by the sample mean graphs. The lone exception was MC, where an epsilon of 0.1 was the fastest to converge.
- *Slowest Convergence*: In three out of four algorithms, the epsilon value of 0.25 had the slowest convergence, as demonstrated by the sample mean graphs. The lone exception was again MC, where 0.01 was the slowest value of epsilon to converge.
- *Lowest Variance*: In the algorithms which had the fastest convergence with an epsilon of 0.01, the same epsilon value also had the lowest variance. MC however, had an epsilon value of 0.1 show the least variance.
- *Highest Variance*: In the algorithms which had the fastest convergence with 0.01, the epsilon value of 0.25 had the highest variance. MC with an epsilon value of 0.01 experienced the highest variance.

If we consider that the epsilon value determines how frequently the agent will take a random action as opposed to the greedy action, it follows logically that the highest value of epsilon to demonstrate the most variance. Since this value is associated with the most variability as the agent is learning, we can expect higher values of variance.

For the three algorithms outside of MC we can see that a epsilon value of 0.01 provided the fastest convergence within the algorithms. Given that a value of 0.01 corresponds to minimal exploration this an expected result. The algorithm is only concerned about taking actions that will maximize its value (essentially it is a greedy policy) and therefore it finds the optimal policy the quickest. However, in the case of TD(Lambda) we see that it arrived at an incorrect policy. This is a consequence of being greedy in that upon increasing value, the algorithm did not explore for potential better policies.

Comparison of Implementation

Which algorithm was most difficult, and which was most easy, to implement and why? Which took the least time to train, and which the most? Which algorithm(s) do you think would scale better to larger problems and why?

Monte Carlo was the most difficult to implement, as it did not use the same framework (bootstrapping) as the other algorithms. Q-Learning and SARSA were relatively similar, with readily-available pseudocode and thus, once the code was written for Q-Learning, SARSA could be written with only minor modifications. TD(Lambda) had the decaying delta vector, which added a bit of complexity. However, once the new pseudocode was shared, the procedure became clear. In theory, Monte Carlo would've been the simplest to implement after TD(lambda) was written, by simply using lambda = 1. However, this led to multiple computational errors (with floating point problems or infinite Q values) so the Monte Carlo Algorithm was written from scratch.

Q-Learning took the fewest iterations and also the shortest time to train. It also produced the best policies, which implies that it would be good for scaling. Part of this is the "aggressive" nature of Q-Learning. The Q values are based on the best case scenario if the optimal action is taken, vs SARSA where the Q values are based on the actual scenario, once the action has been taken according to the e-greedy policy. We already encounter run-time problems (reduced to 10 iterations instead of 20) with Monte Carlo and TD (Lambda) at this scale, in addition to sub-optimal policies, so we would not recommend scaling up with these two.

Appendices

Appendix I: Monte Carlo Q-Values

Average Q-Values over 10 trials

State\Action	U	D	L	R
(0, 1, 1)	-62.7845	-53.3463	-57.6098	-10.2913
(0, 1, 2)	-55.7403	-50.9988	-39.7624	-0.4444
(0, 1, 3)	-100.7709	-4.0717	-81.9416	-64.6096
(0, 1, 4)	-111.9100	-63.9711	-39.2496	-112.1555
(0, 1, 5)	-154.5399	-35.9296	-78.2455	-113.6379
(0, 2, 1)	-29.5018	-68.0358	-62.8373	-65.3027
(0, 2, 2)	-46.4305	-59.3283	-39.9815	-63.4860
(0, 2, 3)	-59.8131	-39.3291	-78.0648	-69.0683
(0, 2, 4)	-60.5521	17.3622	-102.9728	-81.7755
(0, 2, 5)	-67.8163	43.9652	-33.1406	-35.8763
(0, 3, 1)	-57.6880	-37.3771	-31.6401	23.1045
(0, 3, 2)	-35.9218	8.8603	2.2433	-41.9192
(0, 3, 3)	-61.7801	-19.6430	-14.5166	-22.0227
(0, 3, 4)	-30.7964	38.5705	-24.0232	64.5854
(0, 3, 5)	23.6418	77.0094	35.5262	50.9556
(0, 4, 1)	8.8703	-0.9695	-115.8975	-13.2687
(0, 4, 2)	-16.0817	50.2279	-16.7056	28.3946
(0, 4, 3)	11.5032	50.4754	49.1856	75.3391
(0, 4, 4)	67.9079	68.7719	50.3728	82.5177

(0, 5, 1)	17.4248	6.3076	19.7072	63.1612
(0, 5, 2)	42.2204	38.9289	26.3825	59.5400
(0, 5, 3)	66.1168	56.7154	21.0933	61.9313
(0, 5, 4)	75.7896	23.8486	22.7317	31.3204
(0, 5, 5)	70.7434	-103.5185	-99.7326	-70.6437
(1, 1, 1)	13.9967	54.9250	31.3522	22.0311
(1, 1, 2)	15.4640	41.2684	45.2472	40.7576
(1, 1, 3)	21.6566	23.6715	60.2270	19.5482
(1, 1, 4)	18.2162	19.3990	22.5423	-120.0325
(1, 1, 5)	-238.3659	-37.9113	-166.0548	-253.6559
(1, 2, 1)	9.6754	20.6828	17.7993	60.9222
(1, 2, 3)	52.9597	51.4027	74.3055	40.9163
(1, 2, 4)	16.4732	61.5219	7.2371	-14.2278
(1, 2, 5)	-93.1076	42.5713	-50.2694	-152.7088
(1, 3, 1)	2.7819	-143.1492	-10.6212	52.2970
(1, 3, 2)	64.9234	30.6740	14.7355	54.7566
(1, 3, 3)	69.1665	48.8670	62.4520	59.1322
(1, 3, 4)	58.0362	53.7765	59.4509	42.6439
(1, 3, 5)	17.5101	-18.4390	52.1735	-9.9526
(1, 4, 1)	-125.3021	-105.0713	-169.9716	6.3861
(1, 4, 2)	45.3801	-26.3148	1.6565	46.9986
(1, 4, 3)	58.5572	14.8042	44.7792	39.5720
(1, 4, 4)	47.4052	40.2019	48.3859	0.4727
(1, 4, 5)	28.9049	33.0797	41.2143	28.4055

(1, 5, 1)	-63.6273	-57.1578	-56.6139	-7.9671
(1, 5, 2)	21.2036	-36.8690	-22.7816	-40.0258
(1, 5, 3)	36.2011	3.4220	39.5307	-6.7545
(1, 5, 4)	-1.3399	-26.9669	44.5357	-17.0864
(1, 5, 5)	23.7885	-135.7248	3.7884	-37.3671
(2, 1, 1)	-16.3595	29.1777	-20.5446	34.2231
(2, 1, 2)	-0.4223	29.9981	32.4006	12.5956
(2, 1, 3)	-60.5628	21.1516	-31.3258	-88.8754
(2, 1, 4)	-120.8346	-95.4560	-28.1989	-134.0922
(2, 1, 5)	-110.6518	-134.1775	-85.8387	-116.7360
(2, 2, 1)	37.6123	52.6743	34.2510	40.4942
(2, 2, 2)	40.1659	42.6975	49.6504	41.7735
(2, 2, 3)	14.5185	10.9050	42.5052	14.1024
(2, 2, 4)	-36.3679	-8.6679	-33.7885	-74.7010
(2, 2, 5)	-118.8671	-47.7864	-94.6577	-93.7592
(2, 3, 1)	45.2640	56.4607	42.3346	44.8199
(2, 3, 2)	7.1334	40.9699	49.9637	33.6958
(2, 3, 3)	3.9943	-38.6817	42.1319	-64.6179
(2, 3, 4)	-34.6716	-33.9565	-7.3752	-48.4971
(2, 3, 5)	-79.4258	-54.5080	-7.0314	-41.5181
(2, 4, 2)	2.3250	11.3139	53.0871	-30.0581
(2, 4, 3)	-55.2071	-78.7951	20.4944	-61.3913
(2, 4, 4)	-65.5814	-79.6243	-14.9319	-81.4004
(2, 4, 5)	-56.7301	-91.5018	-60.4048	-74.7574

(2, 5, 1)	41.0118	-25.4952	-25.7985	-19.2472
(2, 5, 2)	-28.9487	-30.6619	14.9490	-34.5447
(2, 5, 3)	-41.1514	-29.9608	-32.0408	-55.0627
(2, 5, 4)	-65.3624	-69.7254	-20.8646	-92.2529
(2, 5, 5)	-36.6839	-110.3263	-78.5282	-63.1844
(3, 1, 1)	-26.1773	-9.8108	-20.0237	0.5489
(3, 1, 2)	-7.2515	6.7281	-2.3408	-39.1273
(3, 1, 3)	-6.2144	14.5577	-34.5502	-38.3553
(3, 1, 4)	-16.6101	-12.1114	7.0105	-41.2278
(3, 1, 5)	-41.3782	-3.4496	-8.4529	-38.5154
(3, 2, 1)	-6.3056	13.5724	-1.7179	2.7772
(3, 2, 2)	6.0511	20.6333	0.1652	-0.9764
(3, 2, 3)	-20.1756	-6.4869	7.8141	18.4762
(3, 2, 4)	-9.1683	28.4253	13.6054	6.6045
(3, 2, 5)	-24.2985	-1.2004	25.7853	-7.8468
(3, 3, 1)	15.9260	14.8893	-8.7456	22.3247
(3, 3, 2)	13.7282	18.6465	16.0771	25.8758
(3, 3, 3)	21.7603	24.1768	13.7666	30.7777
(3, 3, 4)	27.5459	36.5887	15.6759	24.3408
(3, 3, 5)	-3.4914	18.7278	23.2114	13.7255
(3, 4, 1)	23.6856	30.3948	22.8413	25.8424
(3, 4, 2)	21.7615	22.6882	20.0062	30.7495
(3, 4, 3)	25.2624	34.0668	25.8725	33.4889
(3, 4, 4)	32.1799	40.7861	32.4865	36.8440

(3, 4, 5)	15.3193	42.0137	32.2029	32.0850
(3, 5, 1)	26.9213	25.6579	22.9100	30.4884
(3, 5, 2)	28.0320	14.6375	24.7798	32.4071
(3, 5, 3)	35.3389	18.1458	21.3222	34.4483
(3, 5, 4)	36.9929	39.5472	35.5435	43.9885
(4, 1, 1)	-14.1607	-2.2931	-2.6515	-1.1496
(4, 1, 2)	-3.9244	-11.9021	-4.5379	-2.1249
(4, 1, 3)	7.4101	16.0336	-8.5604	8.3923
(4, 1, 4)	17.0976	15.2665	12.4142	17.1875
(4, 2, 1)	-17.2839	-13.2038	-13.7286	8.4544
(4, 2, 2)	-0.0481	-5.1628	-3.6028	15.8350
(4, 2, 3)	14.5570	12.9531	10.6499	14.6088
(4, 2, 4)	18.7636	18.8944	18.2984	23.0205
(4, 2, 5)	25.0000	21.1212	21.2685	23.0995
(4, 3, 1)	-13.3140	-18.8598	-1.9478	-1.2544
(4, 3, 2)	-2.1973	4.9114	-3.1402	4.3513
(4, 3, 3)	6.8238	11.4035	10.7149	14.8088
(4, 3, 4)	20.6014	15.4749	15.5388	20.7697
(4, 3, 5)	22.9080	19.4085	19.4479	17.8985
(4, 4, 1)	-15.1702	-2.8485	-16.9829	-5.1420
(4, 4, 2)	1.9625	-1.3374	-18.3151	12.6798
(4, 4, 3)	11.6033	5.5264	3.7976	12.5254
(4, 4, 4)	18.6575	13.1222	12.3929	16.0826
(4, 4, 5)	21.1362	16.7103	17.1675	17.7427

(4, 5, 1)	-5.6715	1.1077	0.4985	3.7246
(4, 5, 2)	10.9219	1.1028	-3.2686	6.5393
(4, 5, 3)	9.7011	4.5115	5.9425	10.0277
(4, 5, 4)	13.3508	3.7564	11.5536	14.8768
(4, 5, 5)	19.2676	16.6477	15.7184	15.1637
(5, 1, 5)	0.0000	0.0000	0.0000	0.0000

Appendix II: Q-Learning Q-Values

Average Q Values over 20 Trials

State\Action	U	D	L	R
(0, 1, 1)	-0.3925	-0.6987	-0.5422	-0.5573
(0, 1, 2)	-0.4531	-3.0037	-0.5632	-0.4096
(0, 1, 3)	-0.1643	-0.2420	-0.3490	-0.3322
(0, 1, 4)	-0.1420	-0.2896	-0.1874	-1.2853
(0, 1, 5)	-0.7519	-0.1330	-0.0940	-0.7574
(0, 2, 1)	-0.6924	-0.7992	-0.8118	-3.0041
(0, 2, 2)	-0.3017	-0.3492	-0.3004	-0.3000
(0, 2, 3)	-0.3064	-0.2185	-3.0000	-0.3269
(0, 2, 4)	-0.1416	-0.3094	-0.3106	-0.2982
(0, 2, 5)	-0.9513	-0.1425	-0.1300	-0.0952
(0, 3, 1)	-0.8458	-3.0001	-0.7938	-0.7671
(0, 3, 2)	-3.0000	14.7898	-0.8079	11.7333
(0, 3, 3)	-0.3269	23.4903	-0.3614	-0.1478

(0, 3, 4)	-0.3066	0.7018	0.8527	21.4685
(0, 3, 5)	-0.0833	52.4295	0.1392	-0.0468
(0, 4, 1)	-0.5521	35.4694	-3.0000	0.3015
(0, 4, 2)	2.5009	18.1903	-0.1432	77.5683
(0, 4, 3)	11.3184	9.3015	21.5023	97.2691
(0, 4, 4)	4.6335	6.1713	7.2323	102.8677
(0, 5, 1)	-3.6921	30.3965	24.8763	94.8739
(0, 5, 2)	77.7747	20.8123	13.3312	26.5210
(0, 5, 3)	52.3202	7.7263	8.6400	11.8394
(0, 5, 4)	33.0206	0.8731	8.0913	-2.6107
(0, 5, 5)	5.7712	-0.2696	-0.2214	-2.9901
(1, 1, 1)	-0.0001	-0.0094	-0.0001	-0.1491
(1, 1, 2)	-0.1648	14.8876	-0.0094	-0.1548
(1, 1, 3)	-0.2814	11.0305	-0.2436	-0.3005
(1, 1, 4)	-0.4936	-0.0926	1.2642	-2.9969
(1, 1, 5)	-1.0749	-0.2177	-0.2888	-1.4970
(1, 2, 1)	-0.1501	-0.0469	-0.1500	0.2443
(1, 2, 3)	1.9069	12.8226	63.4539	3.3418
(1, 2, 4)	-0.5035	-0.2937	36.7665	0.1463
(1, 2, 5)	-3.0296	0.1032	6.8208	-0.7191
(1, 3, 1)	-0.1974	-1.8569	-0.1808	18.7376
(1, 3, 2)	82.8092	16.0972	9.4084	18.6464
(1, 3, 3)	9.2999	12.7006	71.5267	2.3193
(1, 3, 4)	19.4917	-0.6879	14.4078	2.5216

(1, 3, 5)	1.3755	4.2862	12.4169	1.7667
(1, 4, 1)	-0.2613	-0.2341	-1.4824	5.9486
(1, 4, 2)	25.0177	-0.5550	-3.5133	-0.4707
(1, 4, 3)	47.7164	8.7544	20.6007	0.7421
(1, 4, 4)	8.3504	-1.1786	63.9705	23.1412
(1, 4, 5)	10.7015	-4.4044	67.0422	16.5114
(1, 5, 1)	-2.2354	-0.4573	-0.4755	-0.7420
(1, 5, 2)	-0.6006	-0.6396	-0.6984	4.6004
(1, 5, 3)	30.5550	-0.6131	-0.7693	-0.6608
(1, 5, 4)	14.2840	-1.0770	-0.9618	-3.0000
(1, 5, 5)	10.6692	-3.0000	-0.2607	-3.0000
(2, 1, 1)	-0.7019	2.1388	-0.7055	-0.7947
(2, 1, 2)	-0.8620	37.0223	-0.6684	-0.6437
(2, 1, 3)	-0.6066	-0.6084	-0.6573	-0.5950
(2, 1, 4)	-0.5094	-0.6172	-0.5352	-2.9068
(2, 1, 5)	-1.9330	-0.2943	-0.3018	-2.2558
(2, 2, 1)	-0.0942	7.6247	-0.0212	20.4417
(2, 2, 2)	11.2517	56.7283	9.0141	-0.8210
(2, 2, 3)	-0.6213	-0.5782	22.9625	-0.6183
(2, 2, 4)	-0.5096	-0.4026	-0.6139	-0.4255
(2, 2, 5)	-2.7188	-0.4459	-0.3723	-0.2249
(2, 3, 1)	3.8922	64.1350	18.0009	15.4803
(2, 3, 2)	23.1831	9.2434	57.2541	1.7491
(2, 3, 3)	9.6524	-0.3646	-0.1397	-0.3614

(2, 3, 4)	-0.3464	-0.4007	-0.3657	-0.3460
(2, 3, 5)	-0.3432	-0.4451	-0.4205	-0.2624
(2, 4, 2)	0.5743	-0.1343	37.3510	0.1594
(2, 4, 3)	-0.3045	-0.1598	2.5360	-0.2900
(2, 4, 4)	-0.3147	-0.3708	-0.2240	-0.4262
(2, 4, 5)	-0.4384	-2.9765	-0.4589	-0.2136
(2, 5, 1)	0.6241	-0.1594	-0.0103	-0.1785
(2, 5, 2)	-0.0145	-0.1785	-0.1857	-0.0290
(2, 5, 3)	-0.3191	-0.1648	-0.0308	-0.1565
(2, 5, 4)	-0.2326	-0.2514	-0.3174	-1.3827
(2, 5, 5)	-0.3200	-2.8659	-0.1606	-2.8776
(3, 1, 1)	-1.0467	-1.1545	-1.0351	-1.0514
(3, 1, 2)	-1.0155	-1.0151	-1.0351	-1.0677
(3, 1, 3)	-0.8976	-0.9609	-0.9552	-1.0016
(3, 1, 4)	-0.8964	-0.8727	-0.9188	-3.0000
(3, 1, 5)	-0.5512	-0.3116	-0.3000	-2.0446
(3, 2, 1)	-1.1263	-1.2196	-1.2128	-1.1683
(3, 2, 2)	-1.1501	-1.0439	-1.0039	-1.0735
(3, 2, 3)	-0.8867	-0.9602	-0.9739	-0.8065
(3, 2, 4)	-0.6928	-0.7306	-0.6791	-0.6504
(3, 2, 5)	-2.9941	-0.5993	-0.5224	-0.4273
(3, 3, 1)	-1.3103	12.3822	-1.2639	-1.2464
(3, 3, 2)	-1.1198	0.4889	-0.2659	-1.0606
(3, 3, 3)	-0.8778	5.1437	-0.7504	-0.6705

(3, 3, 4)	-0.6784	2.7266	-0.5019	0.6954
(3, 3, 5)	-0.3885	4.5954	-0.3901	-0.3844
(3, 4, 1)	-1.6526	11.6516	12.4170	36.7180
(3, 4, 2)	-1.0951	28.7003	18.2970	17.0809
(3, 4, 3)	0.2692	28.2686	1.5170	3.4085
(3, 4, 4)	-0.1481	6.9092	1.6040	14.9780
(3, 4, 5)	-0.2417	34.3841	-0.1152	-0.2800
(3, 5, 1)	0.4001	-1.0341	3.2913	25.8390
(3, 5, 2)	0.0324	2.9990	2.3445	39.8044
(3, 5, 3)	9.3318	11.3797	14.5859	43.7206
(3, 5, 4)	5.9364	25.7214	23.3057	46.0371
(4, 1, 1)	-0.5583	-0.6329	-0.3340	-0.5728
(4, 1, 2)	-0.5577	-0.4745	-0.3433	-0.0843
(4, 1, 3)	-0.3071	-0.3879	-0.1694	2.2938
(4, 1, 4)	-0.0420	0.2577	-0.0243	20.0873
(4, 2, 1)	-0.7188	-0.6768	-0.6281	-0.6354
(4, 2, 2)	-0.5545	-0.5347	-0.5967	-0.4077
(4, 2, 3)	-0.3794	-0.2842	-0.3662	2.1846
(4, 2, 4)	10.2364	-0.0013	-0.1119	4.2791
(4, 2, 5)	24.9061	2.6757	0.8256	9.8385
(4, 3, 1)	-0.8559	-0.7466	-0.7069	-0.7116
(4, 3, 2)	-0.6824	-0.7259	-0.7577	-0.6399
(4, 3, 3)	-0.4420	-0.5449	-0.5977	-0.6336
(4, 3, 4)	5.2245	-0.6852	-0.6338	6.8476

(4, 3, 5)	23.7109	7.3096	1.8546	2.9716
(4, 4, 1)	-0.8823	-0.8218	-0.7251	-0.7870
(4, 4, 2)	-0.8938	-0.8087	-0.8826	-0.7916
(4, 4, 3)	-0.8428	-0.8341	-0.9492	-0.9636
(4, 4, 4)	2.0461	1.8305	-1.0394	-1.1280
(4, 4, 5)	22.4736	7.3331	-0.7258	12.0574
(4, 5, 1)	-0.8808	-0.9956	-0.9970	-1.0467
(4, 5, 2)	-1.0120	-0.9924	-1.0153	-1.0253
(4, 5, 3)	-1.1102	-1.0042	-1.1227	-1.0657
(4, 5, 4)	-0.8955	-1.4319	-1.4110	8.1807
(4, 5, 5)	21.2495	9.9359	-0.5973	6.9551
(5, 1, 5)	0.0000	0.0000	0.0000	0.0000

Appendix III: SARSA Q-Values

Average Q-Values Over 20 Trials

State\Action	U	D	L	R
(0, 1, 1)	-0.511	-0.318	-0.333	-0.490
(0, 1, 2)	-0.371	-1.496	-0.270	-0.466
(0, 1, 3)	-0.209	-0.420	-0.248	-0.408
(0, 1, 4)	-0.234	-0.435	-0.200	-2.997
(0, 1, 5)	-3.705	-0.291	-0.312	-3.048
(0, 2, 1)	-0.522	-0.520	-0.498	-1.514
(0, 2, 2)	-0.152	-0.312	-0.166	-0.281
(0, 2, 3)	-0.404	-0.428	-2.811	-0.301
(0, 2, 4)	-0.215	-0.244	-0.222	-0.389
(0, 2, 5)	-2.880	-0.272	-0.222	-0.128
(0, 3, 1)	-0.634	-2.999	-0.799	-0.532
(0, 3, 2)	-3.000	7.174	-0.573	2.582
(0, 3, 3)	-0.513	9.794	-0.294	15.195
(0, 3, 4)	-0.227	3.531	-0.262	29.854
(0, 3, 5)	-0.083	57.506	-0.160	-0.079
(0, 4, 1)	-0.573	-0.062	-3.423	9.223
(0, 4, 2)	0.485	1.546	-2.333	21.704
(0, 4, 3)	10.868	0.700	5.247	39.407
(0, 4, 4)	0.209	27.026	4.968	98.935
(0, 5, 1)	-4.333	27.594	31.811	89.664
(0, 5, 2)	17.456	22.019	22.422	74.768

(0, 5, 3)	12.251	34.792	0.898	73.446
(0, 5, 4)	85.330	26.017	22.996	2.168
(0, 5, 5)	40.644	-3.304	-0.066	-1.300
(1, 1, 1)	-0.068	0.000	-0.113	-0.117
(1, 1, 2)	-0.309	11.370	-0.005	-0.148
(1, 1, 3)	-0.245	0.666	0.234	-0.155
(1, 1, 4)	-0.501	-0.332	-0.366	-2.598
(1, 1, 5)	-3.924	-0.285	-0.300	-3.125
(1, 2, 1)	-0.113	-0.150	-0.152	0.256
(1, 2, 3)	-0.143	19.252	72.236	5.408
(1, 2, 4)	-0.461	4.132	36.809	0.583
(1, 2, 5)	-3.000	-0.926	1.810	-0.898
(1, 3, 1)	-0.265	-1.182	-0.068	1.605
(1, 3, 2)	39.751	5.674	0.135	10.081
(1, 3, 3)	50.117	7.737	18.276	9.811
(1, 3, 4)	12.412	1.610	49.467	0.442
(1, 3, 5)	0.148	-0.558	18.051	0.008
(1, 4, 1)	0.058	-0.147	-1.509	-0.153
(1, 4, 2)	16.695	-0.247	-2.935	-0.011
(1, 4, 3)	61.621	2.772	4.333	3.608
(1, 4, 4)	24.889	-0.796	59.131	11.150
(1, 4, 5)	5.975	-4.607	72.024	24.941
(1, 5, 1)	-1.477	-0.326	-0.385	-0.368
(1, 5, 2)	-0.269	-0.360	-0.340	-0.475

(1, 5, 3)	25.565	-0.695	-0.620	0.331
(1, 5, 4)	4.653	-0.951	5.016	-3.000
(1, 5, 5)	5.529	-3.225	-0.594	-4.698
(2, 1, 1)	-0.788	10.881	-0.941	4.086
(2, 1, 2)	-0.851	15.716	-0.461	-0.869
(2, 1, 3)	-0.899	-0.725	-0.906	-0.794
(2, 1, 4)	-0.987	-0.703	-0.849	-3.000
(2, 1, 5)	-3.991	-0.345	-0.311	-0.647
(2, 2, 1)	1.135	58.391	6.216	8.598
(2, 2, 2)	-1.100	12.879	53.580	3.343
(2, 2, 3)	-0.703	0.547	20.975	-0.689
(2, 2, 4)	-0.698	-0.542	-0.345	-0.592
(2, 2, 5)	-2.999	-0.650	-0.643	-0.675
(2, 3, 1)	24.298	63.204	28.539	6.569
(2, 3, 2)	2.421	18.884	12.695	0.361
(2, 3, 3)	-0.477	-0.326	3.916	-0.552
(2, 3, 4)	-0.453	-0.484	-0.494	-0.568
(2, 3, 5)	-0.679	-0.571	-0.557	-0.840
(2, 4, 2)	0.842	-0.230	33.159	-0.099
(2, 4, 3)	-0.387	-0.149	0.472	-0.397
(2, 4, 4)	-0.431	-0.208	-0.346	-0.331
(2, 4, 5)	-0.542	-2.993	-0.300	-0.458
(2, 5, 1)	0.208	-0.001	-0.008	0.000
(2, 5, 2)	0.244	-0.097	-0.008	-0.266

(2, 5, 3)	-0.273	-0.327	-0.299	-0.221
(2, 5, 4)	-0.243	-0.241	-0.368	-1.405
(2, 5, 5)	-0.161	-0.141	-0.290	-1.755
(3, 1, 1)	-1.190	-1.318	-1.299	-1.282
(3, 1, 2)	-1.207	-1.203	-1.181	-1.084
(3, 1, 3)	-1.047	-1.041	-1.131	-0.967
(3, 1, 4)	-0.980	-0.961	-0.999	-3.004
(3, 1, 5)	-3.537	-0.301	-0.322	-1.638
(3, 2, 1)	-1.383	-1.323	-1.441	-1.291
(3, 2, 2)	-1.061	0.024	-1.100	-1.073
(3, 2, 3)	-1.016	-0.868	-0.924	-0.979
(3, 2, 4)	-0.949	-0.700	-0.795	-0.840
(3, 2, 5)	-3.000	-0.704	-0.710	-0.594
(3, 3, 1)	-1.557	-0.156	-1.471	-1.394
(3, 3, 2)	-1.038	12.370	-1.105	-0.438
(3, 3, 3)	-0.909	1.168	-0.877	4.154
(3, 3, 4)	-0.660	6.965	-0.578	-0.410
(3, 3, 5)	-0.572	-0.149	-0.509	-0.584
(3, 4, 1)	-1.651	2.271	2.242	39.508
(3, 4, 2)	1.632	29.401	4.735	12.934
(3, 4, 3)	0.794	4.298	5.140	22.268
(3, 4, 4)	0.912	33.983	4.202	0.482
(3, 4, 5)	-0.177	11.710	-0.298	-0.368
(3, 5, 1)	-1.203	-1.259	-1.274	19.626

(3, 5, 2)	13.075	1.745	3.207	37.024
(3, 5, 3)	2.594	4.831	6.432	43.452
(3, 5, 4)	0.592	16.272	8.604	45.935
(4, 1, 1)	-0.506	-0.480	-0.412	-0.251
(4, 1, 2)	-0.418	-0.351	-0.237	-0.514
(4, 1, 3)	-0.266	-0.317	-0.302	-0.035
(4, 1, 4)	-0.277	-0.011	-0.240	12.049
(4, 2, 1)	-0.427	-0.543	-0.416	-0.523
(4, 2, 2)	-0.494	-0.372	-0.483	-0.432
(4, 2, 3)	-0.330	-0.467	-0.333	0.646
(4, 2, 4)	2.941	-0.139	-0.136	12.093
(4, 2, 5)	25.000	7.967	0.948	1.824
(4, 3, 1)	-0.588	-0.604	-0.418	-0.566
(4, 3, 2)	-0.545	-0.603	-0.604	-0.472
(4, 3, 3)	-0.556	-0.558	-0.585	-0.425
(4, 3, 4)	6.891	-0.476	-0.500	0.041
(4, 3, 5)	23.747	4.227	0.314	8.818
(4, 4, 1)	-0.647	-0.729	-0.885	-0.658
(4, 4, 2)	-0.762	-0.705	-0.658	-0.677
(4, 4, 3)	-0.761	-0.856	-0.716	-0.862
(4, 4, 4)	1.679	-1.074	-0.861	0.331
(4, 4, 5)	22.495	-0.843	-0.666	11.634
(4, 5, 1)	-0.685	-0.797	-0.963	-0.764
(4, 5, 2)	-0.785	-0.787	-0.730	-0.924

(4, 5, 3)	-1.000	-1.045	-0.965	-0.758
(4, 5, 4)	-1.131	-1.351	-1.277	0.461
(4, 5, 5)	21.227	1.432	-1.397	8.278
(5, 1, 5)	0.000	0.000	0.000	0.000

Appendix IV: TD(Lamda) Q-Values

Average Q-Values Over 10 Trials

State\Action	U	D	L	R
(0, 1, 1)	-0.059	-1.755	-2.094	-0.513
(0, 1, 2)	-0.245	-0.293	-0.057	-2.047
(0, 1, 3)	-2.128	-1.783	-1.836	-1.650
(0, 1, 4)	-0.022	-0.386	-1.595	-3.862
(0, 1, 5)	-5.811	-1.811	-1.501	-3.473
(0, 2, 1)	-1.962	-6.707	-2.538	-1.389
(0, 2, 2)	0.069	4.018	-1.685	-2.084
(0, 2, 3)	-1.292	-5.463	0.289	-0.549
(0, 2, 4)	-2.608	0.485	-1.315	4.277
(0, 2, 5)	-2.630	17.972	-0.211	-0.282
(0, 3, 1)	-5.843	-7.954	-8.151	1.105
(0, 3, 2)	-2.452	23.020	0.317	7.774
(0, 3, 3)	-1.793	20.673	-0.625	5.705
(0, 3, 4)	1.384	4.201	0.679	20.097
(0, 3, 5)	12.236	28.126	1.553	24.147
(0, 4, 1)	-5.473	41.838	-6.719	-5.417
(0, 4, 2)	25.364	16.972	4.699	51.606
(0, 4, 3)	16.377	27.132	24.570	68.601
(0, 4, 4)	9.411	19.494	35.607	89.513
(0, 5, 1)	23.562	27.698	16.516	83.222
(0, 5, 2)	52.000	1.180	13.960	52.916

(0, 5, 3)	11.989	15.125	-0.558	70.109
(0, 5, 4)	69.199	64.059	31.096	4.133
(0, 5, 5)	0.899	-0.127	7.252	0.014
(1, 1, 1)	1.746	1.306	1.465	3.570
(1, 1, 2)	5.021	7.016	-0.896	4.063
(1, 1, 3)	8.548	10.864	4.274	0.054
(1, 1, 4)	0.000	6.434	0.028	-0.215
(1, 1, 5)	0.457	0.193	4.904	1.961
(1, 2, 1)	4.294	0.401	18.903	45.907
(1, 2, 3)	8.527	5.762	52.111	16.009
(1, 2, 4)	2.171	9.305	37.773	8.907
(1, 2, 5)	2.026	7.570	18.294	2.672
(1, 3, 1)	44.328	0.214	19.520	0.031
(1, 3, 2)	44.588	7.862	43.405	2.906
(1, 3, 3)	26.007	39.742	31.941	9.852
(1, 3, 4)	30.764	18.856	53.901	14.152
(1, 3, 5)	16.245	20.413	48.529	9.335
(1, 4, 1)	1.132	2.055	-0.102	4.416
(1, 4, 2)	46.797	13.236	-1.954	8.883
(1, 4, 3)	35.963	10.240	34.651	-2.750
(1, 4, 4)	51.819	11.542	5.907	4.847
(1, 4, 5)	44.266	7.202	47.221	27.967
(1, 5, 1)	3.251	0.865	1.196	0.148
(1, 5, 2)	-0.288	6.001	1.215	16.108

(1, 5, 3)	9.679	4.925	1.277	22.998
(1, 5, 4)	30.114	-2.666	17.011	-5.462
(1, 5, 5)	12.174	-5.998	4.772	2.123
(2, 1, 1)	3.452	5.525	0.605	5.301
(2, 1, 2)	36.192	24.653	8.554	-0.452
(2, 1, 3)	4.843	-0.336	6.550	-2.083
(2, 1, 4)	-2.295	-1.352	-1.471	-2.805
(2, 1, 5)	-0.426	-0.399	-1.532	-0.014
(2, 2, 1)	0.930	6.142	-1.382	15.663
(2, 2, 2)	19.558	33.289	4.882	30.049
(2, 2, 3)	3.785	28.893	12.591	-1.367
(2, 2, 4)	-1.953	0.602	-1.121	0.533
(2, 2, 5)	-2.917	0.194	0.483	-0.473
(2, 3, 1)	13.641	50.326	2.422	20.638
(2, 3, 2)	21.436	16.399	41.446	13.628
(2, 3, 3)	13.006	26.801	33.075	1.452
(2, 3, 4)	-0.080	1.990	0.319	0.601
(2, 3, 5)	-2.731	0.573	0.676	-2.514
(2, 4, 2)	3.697	1.226	43.917	3.804
(2, 4, 3)	0.027	-2.165	30.840	0.043
(2, 4, 4)	0.129	-2.512	6.696	-5.739
(2, 4, 5)	-2.457	-6.571	1.451	1.093
(2, 5, 1)	1.142	0.015	-3.915	-1.909
(2, 5, 2)	2.274	-6.206	-0.785	1.092

(2, 5, 3)	-0.310	-4.029	3.224	-0.913
(2, 5, 4)	-4.715	-9.441	2.558	-5.518
(2, 5, 5)	-2.337	-7.745	-3.169	-9.249
(3, 1, 1)	-4.086	-2.817	-2.738	-2.725
(3, 1, 2)	-3.847	-3.834	-3.091	-2.357
(3, 1, 3)	-1.629	-1.664	-2.590	-2.673
(3, 1, 4)	-3.389	-1.821	-1.661	-1.450
(3, 1, 5)	-2.408	0.501	-0.403	-1.515
(3, 2, 1)	-2.750	-1.602	-2.897	-2.920
(3, 2, 2)	-3.949	-0.474	-4.169	-2.316
(3, 2, 3)	-2.940	4.412	-2.337	-1.955
(3, 2, 4)	-1.718	-2.259	1.648	0.534
(3, 2, 5)	-0.394	2.967	-0.324	0.012
(3, 3, 1)	-1.043	0.693	-2.337	4.418
(3, 3, 2)	-1.479	5.949	-1.267	16.909
(3, 3, 3)	2.642	13.020	0.825	8.618
(3, 3, 4)	2.303	9.341	2.260	5.901
(3, 3, 5)	1.567	8.210	5.162	2.800
(3, 4, 1)	3.800	23.252	16.414	32.135
(3, 4, 2)	12.332	19.784	10.097	22.762
(3, 4, 3)	1.940	22.586	1.843	11.534
(3, 4, 4)	5.876	1.137	-1.418	14.823
(3, 4, 5)	6.879	25.795	2.864	14.412
(3, 5, 1)	2.643	1.302	-3.439	20.617

(3, 5, 2)	11.466	4.745	-1.304	20.937
(3, 5, 3)	5.188	0.250	8.187	28.112
(3, 5, 4)	10.391	26.775	11.680	24.483
(4, 1, 1)	-1.106	-1.182	-1.050	-0.149
(4, 1, 2)	-1.227	-1.436	-1.314	0.988
(4, 1, 3)	3.690	6.681	-0.977	20.366
(4, 1, 4)	11.950	6.352	7.395	23.082
(4, 2, 1)	-1.384	-0.738	-1.308	-1.074
(4, 2, 2)	-1.382	-0.481	-1.269	5.665
(4, 2, 3)	16.782	1.070	-1.150	1.809
(4, 2, 4)	12.342	-0.263	3.880	6.568
(4, 2, 5)	15.411	-0.738	3.067	2.471
(4, 3, 1)	-1.118	1.805	-0.281	5.603
(4, 3, 2)	3.804	-1.378	2.838	-2.154
(4, 3, 3)	10.108	0.483	-0.299	-0.206
(4, 3, 4)	15.774	-0.590	2.181	13.176
(4, 3, 5)	10.964	9.540	12.912	11.489
(4, 4, 1)	2.252	-1.902	-2.054	-1.197
(4, 4, 2)	-2.481	-1.137	-1.895	-0.879
(4, 4, 3)	6.237	0.294	-1.930	0.735
(4, 4, 4)	3.695	-3.260	1.227	5.095
(4, 4, 5)	11.759	-0.022	3.099	-4.625
(4, 5, 1)	-1.984	-2.052	-2.692	-2.474
(4, 5, 2)	-2.885	-3.553	-3.012	-0.664

(4, 5, 3)	4.271	-5.238	-3.078	-2.732
(4, 5, 4)	1.573	-6.308	-1.108	-3.422
(4, 5, 5)	14.208	2.180	1.828	-5.121
(5, 1, 5)	0.000	0.000	0.000	0.000