Wrocław University of Science and Technology

Faculty of Electronics, Photonics and Microsystems

Artificial Intelligence and Machine Learning

Wednesday 15:15 – 16:55

Markov decision problems and reinforcement learning

Author:

Maciej Kaniewski, 254011

Tutor:

Ph.D. Witold Paluszyński

May 22, 2023

Wrocław University of Science and Technology

1 MDP results for the Russell and Norvig 4x3 world

	1	2	3	4
3	> 0.8116	> 0.8678	> 0.9178	T 1.0000
2	^ 0.7616	F 0.0000	^ 0.6603	T -1.0000
1	S ^ 0.7053	< 0.6553	< 0.6114	< 0.3879

Figure 1.1: Utilities and policy for the 4x3 problem presented in the lecture.

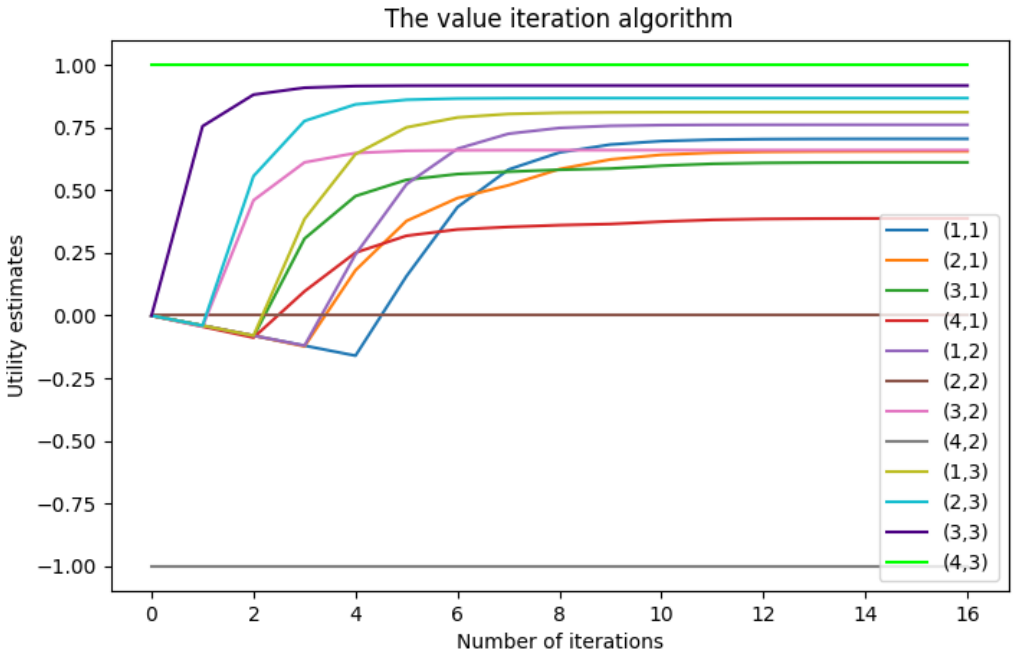

Figure 1.2: Convergence plot for the 4x3 problem presented in the lecture.

2 MDP results for the 4x4 basic world

4	> 81.9384	> 84.2610	> 86.5861	v 88.8827
3	> 81.7354	> 84.2724	> 87.0596	v 91.5547
2	^ 79.5936	^ 80.5997	B > 70.4670	v 94.5352
1	S ^ 77.4526	^ 78.2495	F 0.0000	T 100.0000
	1	2	3	4

Figure 2.1: Utilities and policy for the 4x4 basic world.

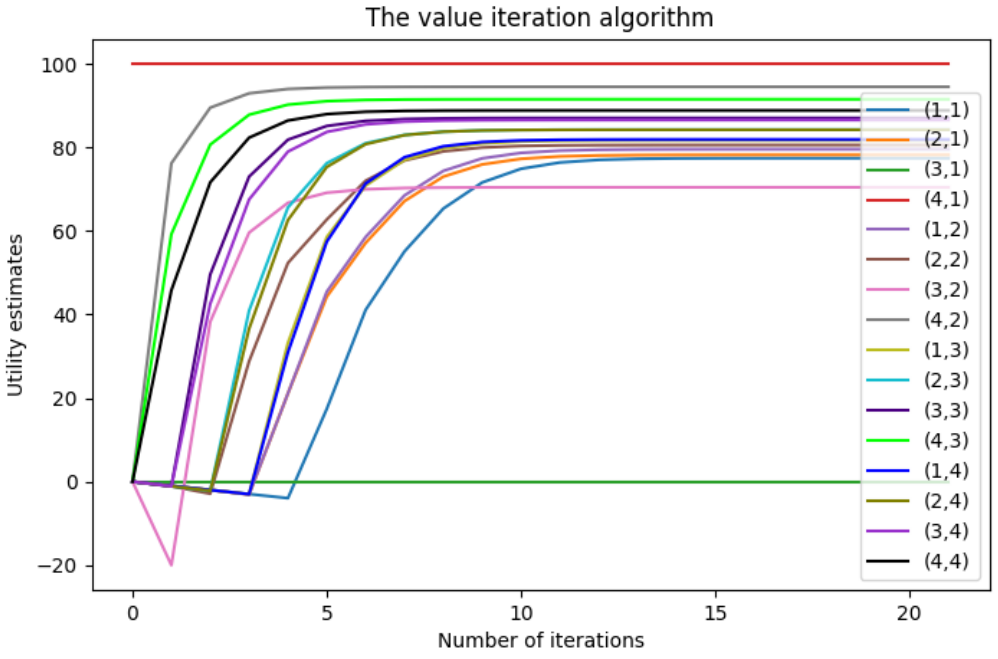

Figure 2.2: Convergence plot for the 4x4 basic world.

3 MDP results for the 4x4 basic world modifications

3.1 Modified reward function

It was observed that for the value of the reward of special state in the range $(-6.2, 42.5)$ the policy of moves does not differ from the world with default parameters. By setting the value to -6.2, a policy change was observed in the states `(1,2)` and `(2,2)`, which suggests entering a special state 3.1. This is due to the fact that the penalty for entering this state has been reduced, thus there is a shorter path to the final state through it. By setting the reward in the special state to -42.5, a policy change in the state `(3,3)` was observed, which suggests going up to minimize the chance of landing in a special state with a significantly increased penalty 3.3.

3.2 Modified uncertainty model

In this modification, the probability distribution of the agent's moves has been changed. They were changed to $p_1 = 0.2(\hat{})$, $p_2 = 0.1(<)$ and $p_3 = 0.1(>)$ respectively. Thus, it has been observed that the resulting move policy is completely opposite 3.5 to that obtained with default values. This is due to the fact that by setting the probability in this way, the chance of transferring the agent in the opposite state to the intended one is 60%, because it is equal to $1 - p_1 - p_2 - p_3$. The solution converges after a much larger number of iterations of the algorithm 3.6.

3.3 Modified discounting

The last modification was the change of the gamma parameter. Here, it was observed that by changing this parameter from 0.99 to 0.90, the utility value of all states decreases. With a gamma parameter of 0.90, a policy change was observed in states `(1,2)` and `(2,2)` suggesting a shift towards a special state with a negative reward value 3.7. This is because by decreasing the value of the gamma parameter, the weight of rewards resulting from future states is reduced. This also translates into a smaller number of iterations of the algorithm 3.8.

3.4 Modifications results

4	> 84.3626	> 86.5929	> 88.7949	v 90.9478
3	> 85.1423	> 87.8244	> 90.6418	v 93.6279
2	> 83.1661	> 85.4889	B > 87.8565	v 96.4459
1	S ^ 81.1093	^ 82.9490	F 0.0000	T 100.0000
	1	2	3	4

Figure 3.1: Utilities and policy for the special state reward modification to -6.2.

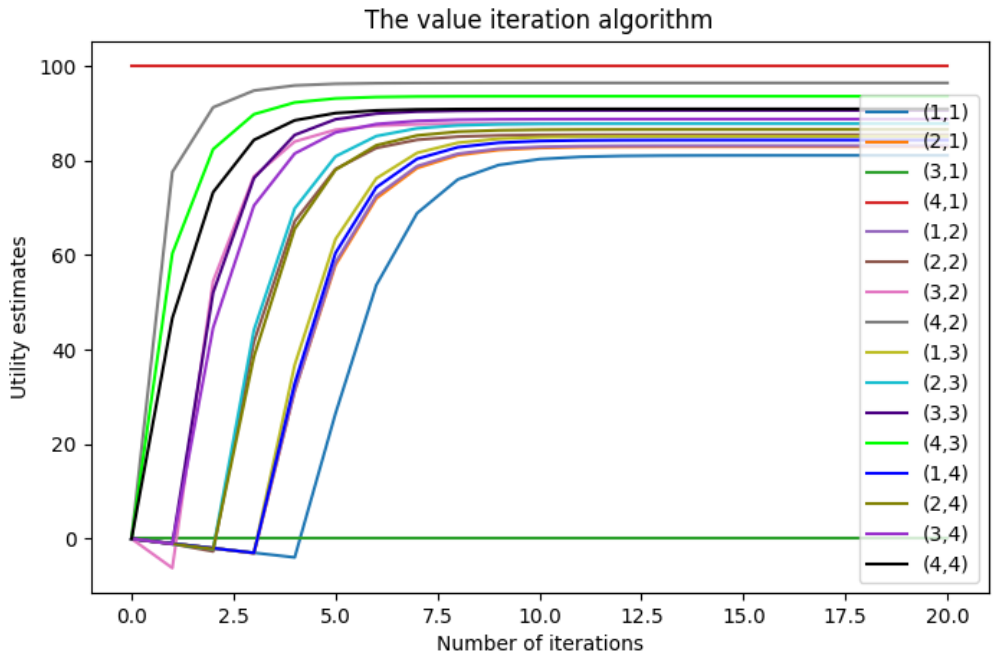

Figure 3.2: Convergence plot for the special state reward modification to -6.2.

	1	2	3	4
4	> 77.9888	> 80.4606	> 82.9859	v 85.5165
3	> 76.1934	> 78.4868	^ 81.2243	v 88.1752
2	^ 73.8476	^ 72.6419	B > 42.1153	v 91.4200
1	S ^ 71.5620	^ 70.6071	F 0.0000	T 100.0000

Figure 3.3: Utilities and policy for the special state reward modification to -42.5.

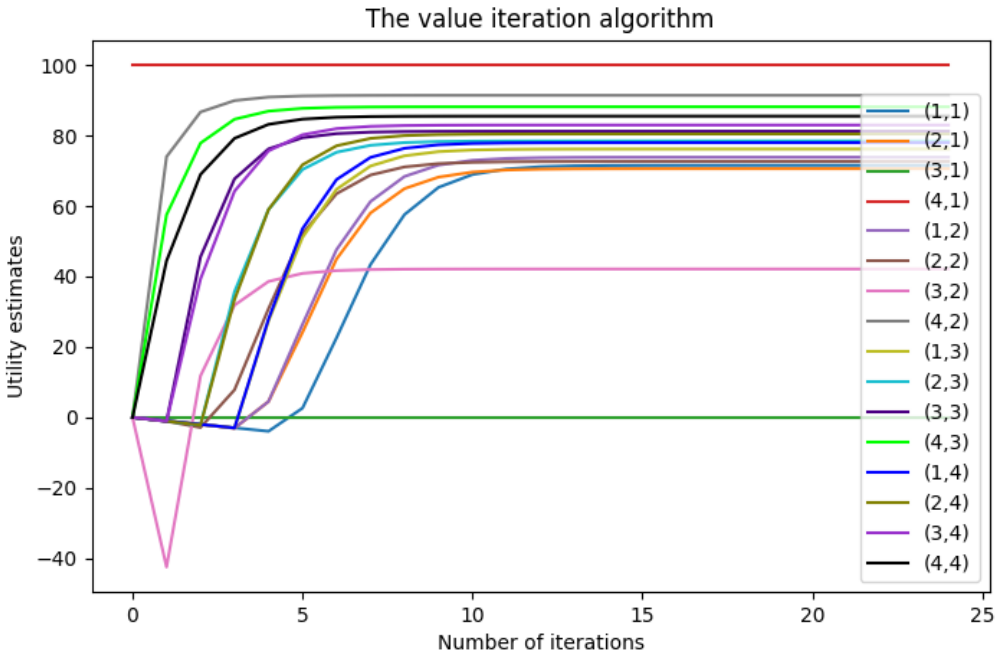

Figure 3.4: Convergence plot for the special state reward modification to -42.5.

4	< 69.5382	< 72.5045	< 76.4546	^ 80.6576
3	< 68.8660	< 72.1611	< 76.9609	^ 84.3995
2	v 65.4812	v 66.9988	B < 60.3102	^ 89.9909
1	S v 62.9256	v 64.0496	F 0.0000	T 100.0000
	1	2	3	4

Figure 3.5: Utilities and policy for the uncertainty model modification.

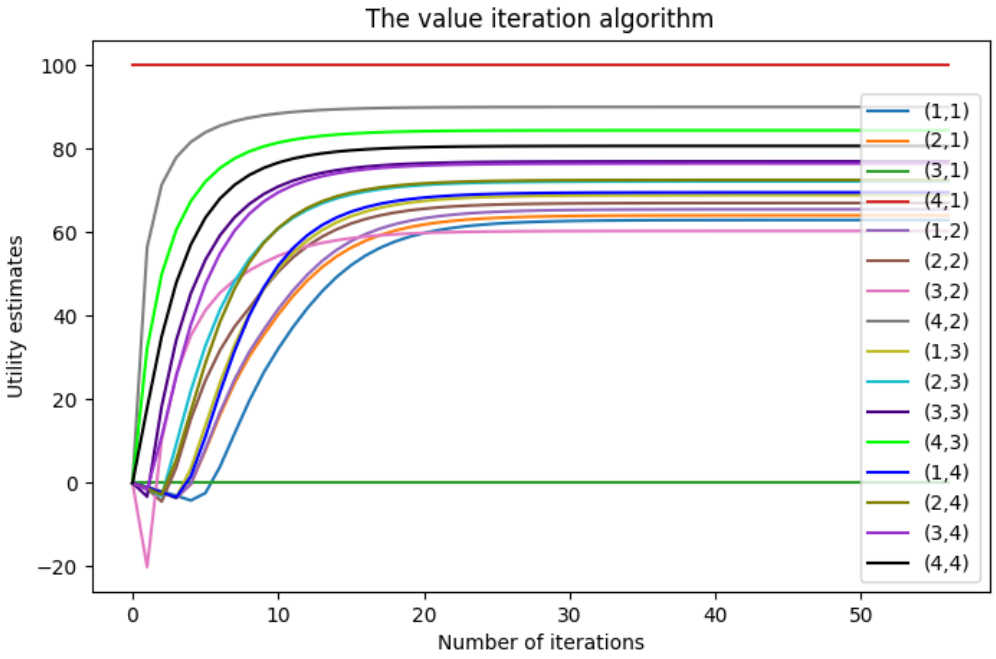

Figure 3.6: Convergence plot for the uncertainty model modification.

	1	2	3	4
4	> 38.4392	> 44.8395	> 51.9198	v 59.6666
3	> 41.0585	> 49.1296	> 58.7455	v 70.3109
2	> 35.8411	> 42.1918	B > 49.4318	v 82.9108
1	S ^ 30.7525	^ 35.3251	F 0.0000	T 100.0000

Figure 3.7: Utilities and policy for the discounting modification.

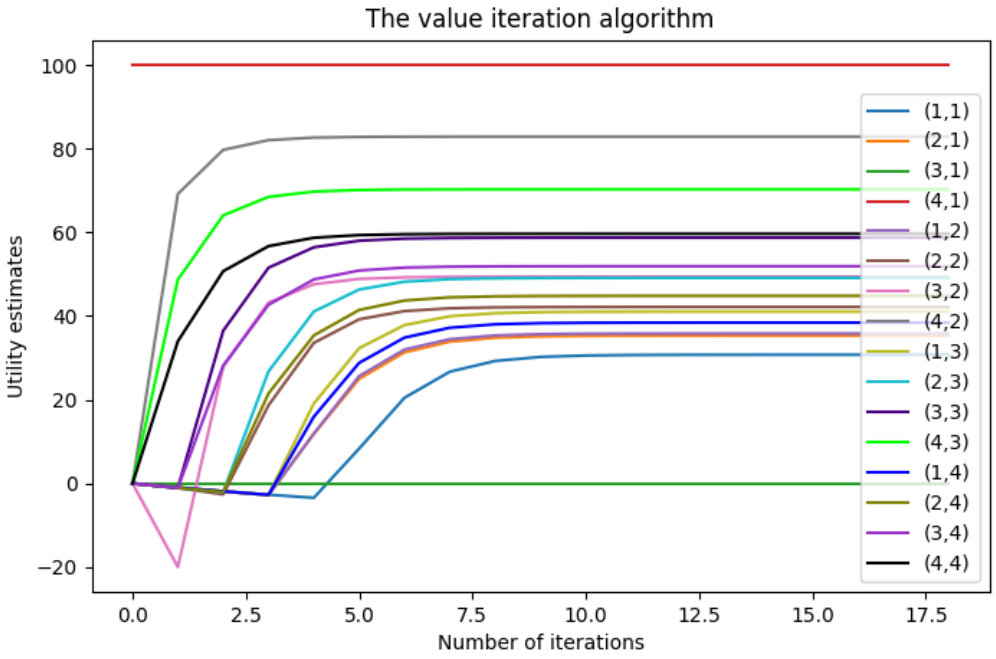

Figure 3.8: Convergence plot for the discounting modification.

4 Q-learning results for the 4x4 basic world

4.1 $\varepsilon = 0.05, iterations = 10000$

	1	2	3	4
4	> 24.4875	> 42.5999	> 62.0601	v 75.7082
3	> 43.3055	> 62.8063	> 77.3232	v 87.8734
2	^ 30.7171	> 60.9135	B > 67.4468	v 93.9901
1	S > 38.7502	^ 51.2824	F 0.0000	T 100.0000

Figure 4.1: Utilities and policy for the $\varepsilon = 0.05$, $i = 10000$.

	1	2	3	4
^	-4.1705	1.2704	20.7413	18.5140
<	-4.2297	-1.0014	4.3133	36.4544
>	24.4875	42.5999	62.0601	34.7935
v	1.7707	11.7226	29.6340	75.7082
^	4.8112	15.9121	21.2344	56.4839
<	4.0764	13.0531	45.0564	59.0287
>	43.3055	62.8063	77.3232	74.0885
v	8.9580	36.5510	60.8496	87.8734
^	30.7171	30.3801	49.7557	78.0107
<	0.8107	13.6238	38.1203	68.0467
>	11.8369	60.9135	67.4468	88.6096
v	6.5176	42.1868	44.2258	93.9901
^	13.6332	51.2824	0.0000	100.0000
<	13.3397	26.5155	0.0000	100.0000
>	38.7502	26.3042	0.0000	100.0000
v	14.5135	26.0215	0.0000	100.0000

Figure 4.2: Q values for the $\varepsilon = 0.05$, $i = 10000$.

4.2 $\varepsilon = 0.05, iterations = 100000$

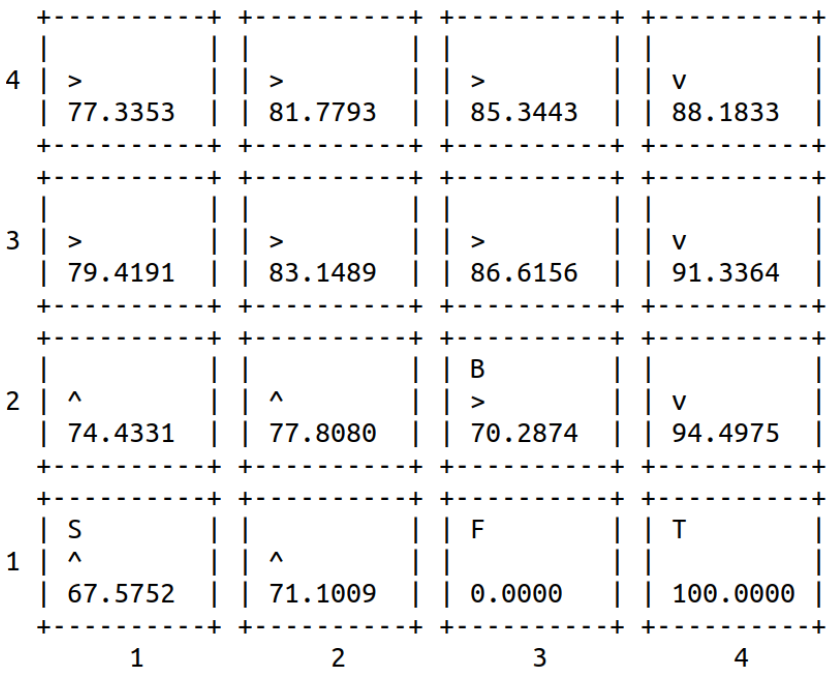

Figure 4.3: Utilities and policy for the $\varepsilon = 0.05$, $i = 100000$.

^ 74.0616 < 72.7998 > 77.3353 v 72.0560	^ 78.2233 < 74.5990 > 81.7792 v 77.4230	^ 82.7286 < 79.7862 > 85.3443 v 83.6822	^ 85.4660 < 83.5129 > 86.0863 v 88.1833
^ 73.0838 < 74.1266 > 79.4191 v 70.1429	^ 77.5108 < 76.1944 > 83.1489 v 75.2869	^ 83.5128 < 79.6919 > 86.6156 v 71.8595	^ 86.1453 < 85.2972 > 89.0216 v 91.3364
^ 74.4331 < 68.5380 > 72.2421 v 63.1279	^ 77.8080 < 69.5130 > 69.2506 v 65.6667	^ 65.0097 < 54.5799 > 70.2874 v 51.9447	^ 87.4792 < 73.6199 > 92.5041 v 94.4975
^ 67.5752 < 61.8725 > 64.4992 v 61.1568	^ 71.1009 < 60.7993 > 62.3094 v 61.7696	^ 0.0000 < 0.0000 > 0.0000 v 0.0000	^ 100.0000 < 100.0000 > 100.0000 v 100.0000

Figure 4.4: Q values for the $\varepsilon = 0.05$, $i = 100000$.

4.3 $\varepsilon = 0.05, iterations = 500000$

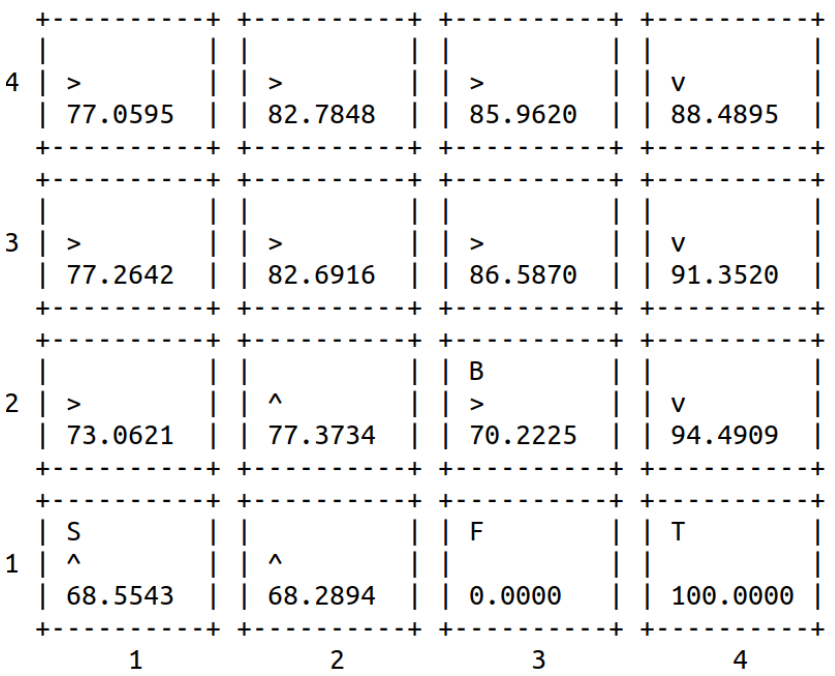

Figure 4.5: Utilities and policy for the $\varepsilon = 0.05$, $i = 500000$.

^ 70.0905	^ 78.4365	^ 83.2070	^ 79.9733
< 70.4042	< 71.8738	< 80.3833	< 84.0049
> 77.0595	> 82.7848	> 85.9620	> 86.4313
v 71.5209	v 69.0511	v 81.6708	v 88.4895
^ 70.9282	^ 79.0860	^ 83.4992	^ 86.2477
< 70.9971	< 73.0086	< 78.6865	< 85.2296
> 77.2642	> 82.6916	> 86.5870	> 88.4977
v 69.5283	v 74.3384	v 70.4840	v 91.3520
^ 65.1757	^ 77.3734	^ 64.8670	^ 87.4248
< 69.1942	< 69.0740	< 54.8338	< 73.2496
> 73.0621	> 68.8761	> 70.2224	> 92.6943
v 66.1118	v 63.6557	v 52.3695	v 94.4909
^ 68.5543	^ 68.2894	^ 0.0000	^ 100.0000
< 64.2514	< 58.8190	< 0.0000	< 100.0000
> 51.6166	> 60.2816	> 0.0000	> 100.0000
v 63.4690	v 59.4609	v 0.0000	v 100.0000

Figure 4.6: Q values for the $\varepsilon = 0.05$, $i = 500000$.

4.4 $\varepsilon = 0.05, iterations = 2500000$

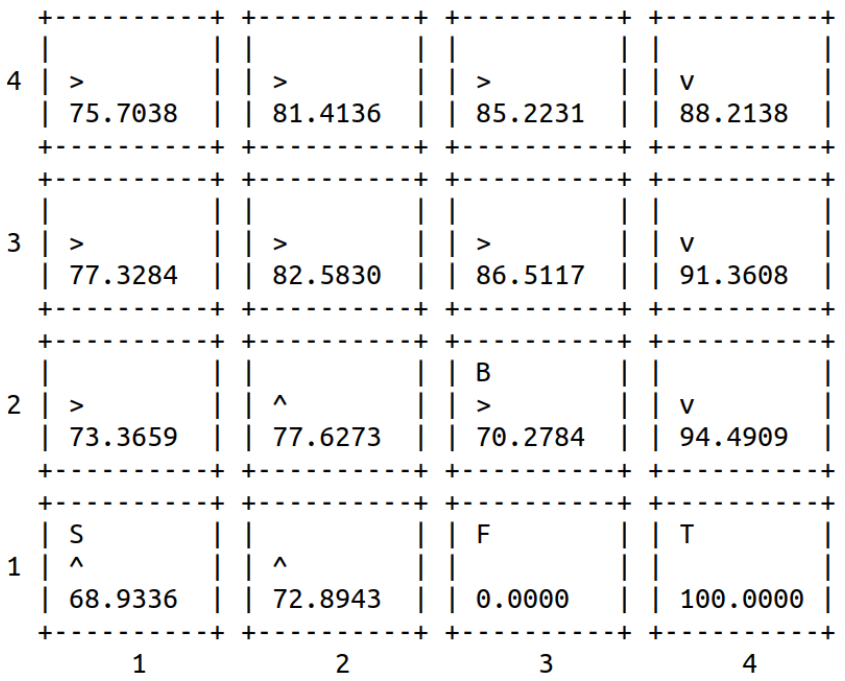

Figure 4.7: Utilities and policy for the $\varepsilon = 0.05$, $i = 2500000$.

^ 69.5851	^ 76.8198	^ 82.0154	^ 85.3790
< 68.8221	< 71.2136	< 78.5159	< 83.2904
> 75.7038	> 81.4136	> 85.2231	> 86.0414
v 72.1276	v 78.6528	v 83.5787	v 88.2138
^ 70.0639	^ 76.3646	^ 82.7012	^ 85.9635
< 71.4774	< 73.4455	< 78.8684	< 85.2163
> 77.3284	> 82.5830	> 86.5117	> 89.2519
v 69.9313	v 75.2703	v 71.5729	v 91.3608
^ 66.5681	^ 77.6274	^ 64.9930	^ 87.5576
< 69.2872	< 70.3571	< 55.8669	< 73.4271
> 73.3659	> 69.6495	> 70.2784	> 92.7370
v 65.7532	v 68.9293	v 52.4572	v 94.4909
^ 68.9336	^ 72.8943	^ 0.0000	^ 100.0000
< 64.7177	< 65.5762	< 0.0000	< 100.0000
> 68.3400	> 68.8016	> 0.0000	> 100.0000
v 64.5789	v 68.0773	v 0.0000	v 100.0000

Figure 4.8: Q values for the $\varepsilon = 0.05$, $i = 2500000$.

4.5 $\varepsilon = 0.05, iterations = 5000000$

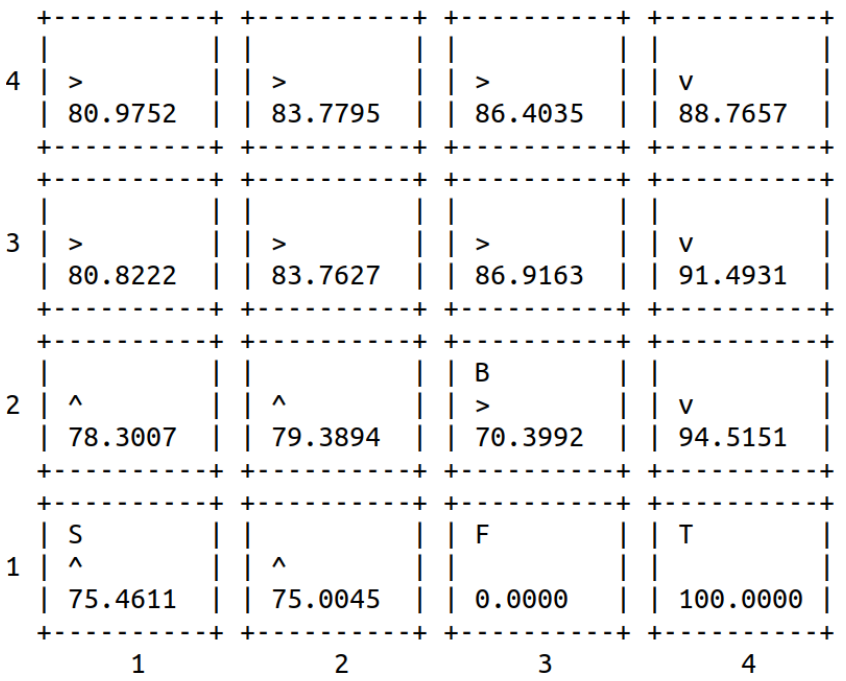

Figure 4.9: Utilities and policy for the $\varepsilon = 0.05$, $i = 5000000$.

^ 78.8682	^ 81.3646	^ 84.2699	^ 86.5661
< 78.5968	< 78.8287	< 82.0183	< 84.5642
> 80.9752	> 83.7795	> 86.4035	> 87.0769
v 78.9273	v 80.6112	v 83.8162	v 88.7657
^ 78.8957	^ 81.3706	^ 84.5202	^ 86.8102
< 78.4218	< 78.4978	< 80.5432	< 85.8734
> 80.8222	> 83.7627	> 86.9163	> 89.5598
v 76.5945	v 77.8793	v 72.0434	v 91.4931
^ 78.3007	^ 79.3894	^ 65.8327	^ 87.7798
< 75.9568	< 74.9584	< 57.8643	< 73.6632
> 74.9811	> 70.1733	> 70.3992	> 92.8066
v 73.6788	v 71.6622	v 52.5725	v 94.5151
^ 75.4611	^ 75.0045	^ 0.0000	^ 100.0000
< 72.7112	< 70.3983	< 0.0000	< 100.0000
> 68.9103	> 71.5676	> 0.0000	> 100.0000
v 72.3445	v 70.9052	v 0.0000	v 100.0000

Figure 4.10: Q values for the $\varepsilon = 0.05$, $i = 5000000$.

4.6 $\varepsilon = 0.2, iterations = 10000$

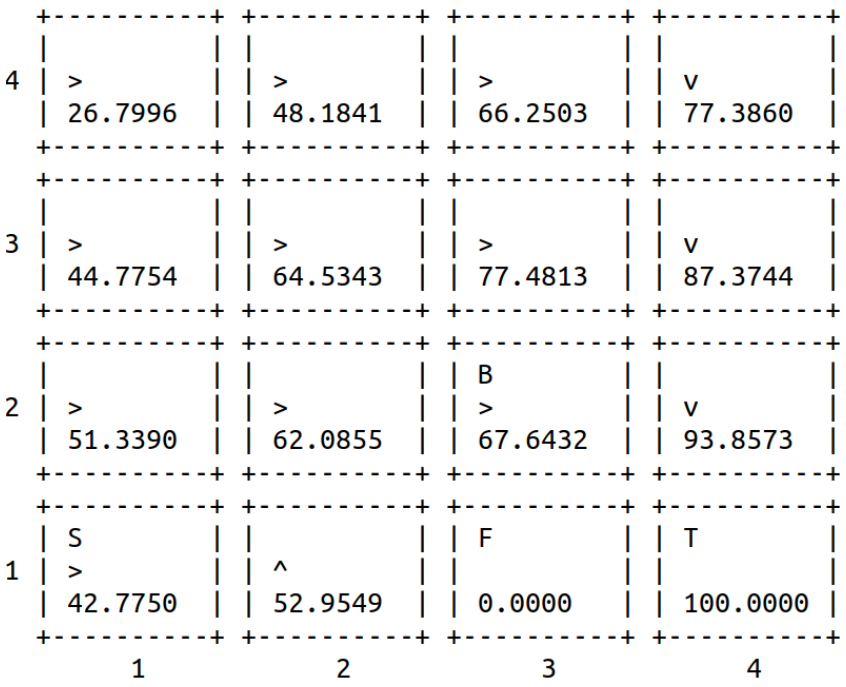

Figure 4.11: Utilities and policy for the $\varepsilon = 0.2, i = 10000$.

^ 8.8803	^ 18.7704	^ 52.1235	^ 65.8067
< 7.3566	< 19.3679	< 34.6082	< 59.6172
> 26.7996	> 48.1841	> 66.2503	> 64.0446
v 17.3493	v 33.4232	v 59.6680	v 77.3860
^ 14.1290	^ 28.5997	^ 54.3103	^ 71.0372
< 19.3100	< 32.6072	< 54.4014	< 72.4761
> 44.7754	> 64.5343	> 77.4813	> 81.4977
v 31.7959	v 48.2418	v 61.7180	v 87.3744
^ 21.6734	^ 47.9657	^ 50.4962	^ 81.3289
< 37.0020	< 43.2042	< 40.6049	< 68.7933
> 51.3390	> 62.0855	> 67.6432	> 90.7070
v 31.9985	v 44.9419	v 46.6974	v 93.8573
^ 39.2641	^ 52.9549	^ 0.0000	^ 100.0000
< 30.5654	< 35.4587	< 0.0000	< 100.0000
> 42.7750	> 44.1295	> 0.0000	> 100.0000
v 30.3079	v 41.9581	v 0.0000	v 100.0000

Figure 4.12: Q values for the $\varepsilon = 0.2, i = 10000$.

4.7 $\varepsilon = 0.2, iterations = 100000$

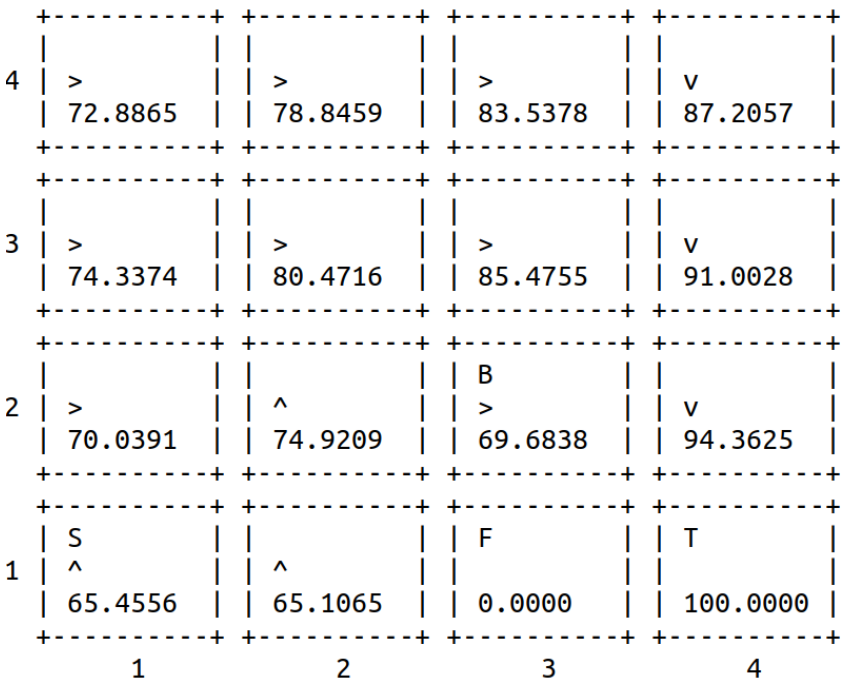

Figure 4.13: Utilities and policy for the $\varepsilon = 0.2$, $i = 100000$.

^ 64.2259	^ 71.4980	^ 77.8135	^ 83.0256
< 64.3650	< 67.1889	< 74.0867	< 78.3313
> 72.8865	> 78.8459	> 83.5378	> 84.0148
v 61.3457	v 74.2157	v 81.7875	v 87.2057
^ 66.1645	^ 72.6968	^ 78.2579	^ 84.1149
< 67.3559	< 68.9515	< 75.5515	< 83.5584
> 74.3374	> 80.4716	> 85.4755	> 88.5547
v 65.7787	v 71.7521	v 70.6899	v 91.0028
^ 69.0369	^ 74.9209	^ 62.0669	^ 86.7252
< 64.9125	< 63.6148	< 50.3401	< 72.3906
> 70.0391	> 65.8816	> 69.6838	> 92.2627
v 61.7109	v 61.2638	v 50.7937	v 94.3625
^ 65.4556	^ 65.1065	^ 0.0000	^ 100.0000
< 57.4883	< 55.2552	< 0.0000	< 100.0000
> 57.3175	> 59.3826	> 0.0000	> 100.0000
v 57.1374	v 57.9243	v 0.0000	v 100.0000

Figure 4.14: Q values for the $\varepsilon = 0.2$, $i = 100000$.

4.8 $\varepsilon = 0.2, iterations = 500000$

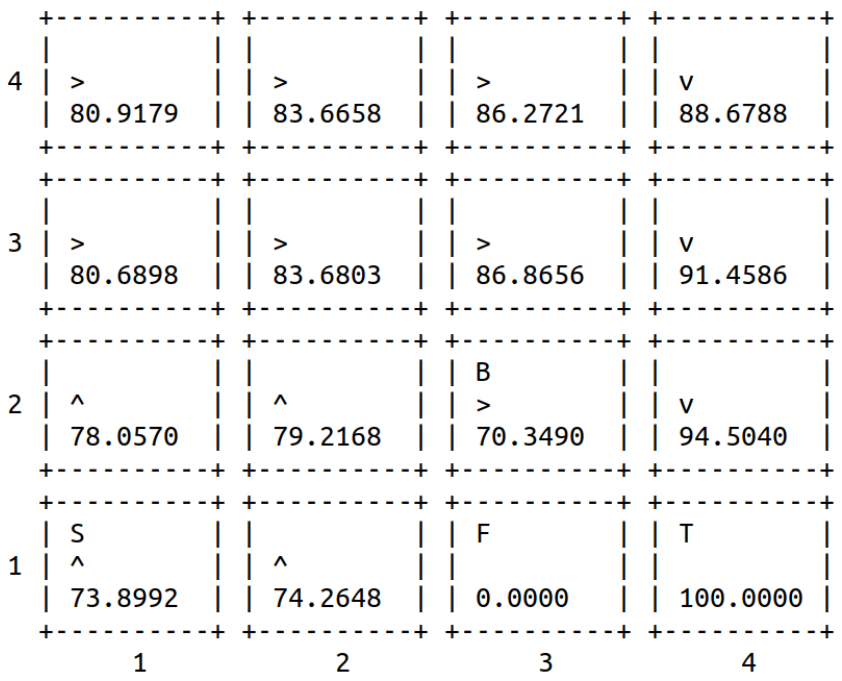

Figure 4.15: Utilities and policy for the $\varepsilon = 0.2$, $i = 500000$.

^ 78.6373	^ 81.2995	^ 84.2080	^ 86.4461
< 78.4651	< 78.8489	< 81.9664	< 84.9751
> 80.9179	> 83.6658	> 86.2721	> 86.9575
v 77.8262	v 81.3069	v 84.3765	v 88.6788
^ 78.8308	^ 81.2834	^ 84.4335	^ 86.7562
< 78.0913	< 78.0590	< 80.3789	< 85.7996
> 80.6898	> 83.6803	> 86.8656	> 89.4961
v 76.4040	v 77.6029	v 71.8573	v 91.4586
^ 78.0570	^ 79.2168	^ 65.7303	^ 87.6691
< 74.6925	< 73.7316	< 57.3925	< 73.5433
> 74.7737	> 68.7609	> 70.3490	> 92.7548
v 71.0974	v 69.1667	v 52.5897	v 94.5040
^ 73.8992	^ 74.2648	^ 0.0000	^ 100.0000
< 70.0751	< 68.7482	< 0.0000	< 100.0000
> 66.4398	> 69.2822	> 0.0000	> 100.0000
v 69.5916	v 68.4626	v 0.0000	v 100.0000

Figure 4.16: Q values for the $\varepsilon = 0.2$, $i = 500000$.

4.9 $\varepsilon = 0.2, iterations = 2500000$

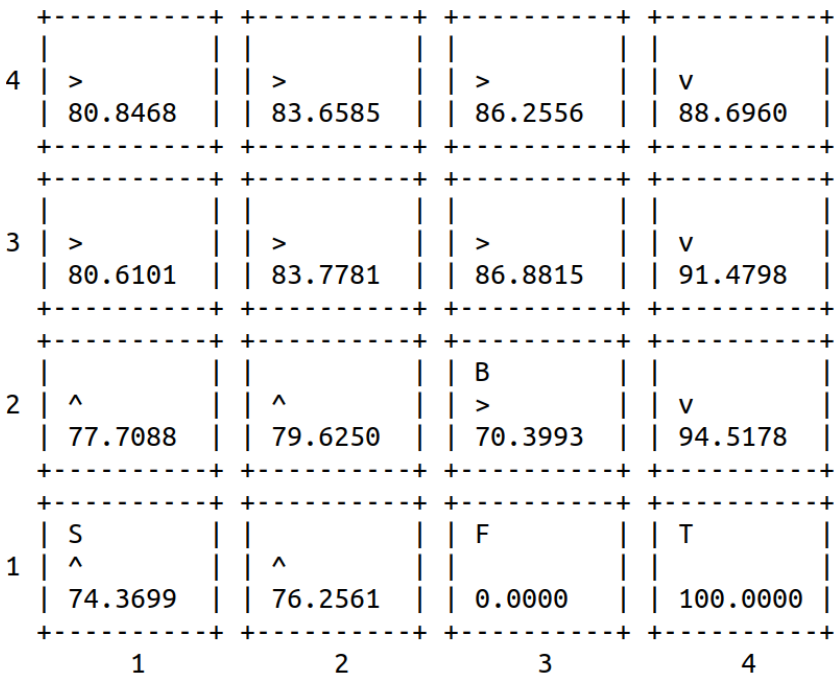

Figure 4.17: Utilities and policy for the $\varepsilon = 0.2, i = 2500000$.

^ 78.6868	^ 81.4084	^ 84.1640	^ 86.4303
< 78.3549	< 79.0510	< 82.0547	< 84.9662
> 80.8468	> 83.6585	> 86.2556	> 86.9624
v 77.7325	v 81.5430	v 84.6549	v 88.6960
^ 78.6933	^ 81.4531	^ 84.4542	^ 86.7842
< 77.8180	< 78.3648	< 80.5536	< 85.8375
> 80.6101	> 83.7781	> 86.8815	> 89.5319
v 75.8892	v 78.1027	v 72.0172	v 91.4797
^ 77.7088	^ 79.6250	^ 65.9127	^ 87.7117
< 74.8286	< 75.4394	< 58.1511	< 73.6275
> 76.7191	> 70.3764	> 70.3993	> 92.7791
v 72.1146	v 73.1007	v 52.8454	v 94.5178
^ 74.3699	^ 76.2561	^ 0.0000	^ 100.0000
< 71.4850	< 71.9332	< 0.0000	< 100.0000
> 73.3639	> 73.7643	> 0.0000	> 100.0000
v 71.3472	v 73.2041	v 0.0000	v 100.0000

Figure 4.18: Q values for the $\varepsilon = 0.2, i = 2500000$.

4.10 $\varepsilon = 0.2, iterations = 5000000$

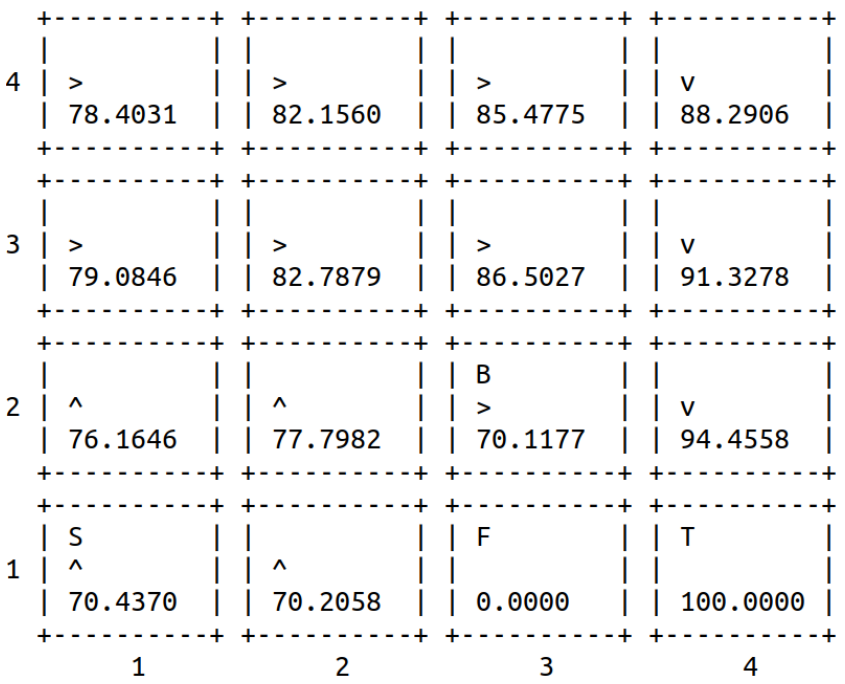

Figure 4.19: Utilities and policy for the $\varepsilon = 0.2$, $i = 5000000$.

^ 74.4952	^ 78.5539	^ 82.6388	^ 85.6164
< 74.0616	< 74.2107	< 79.5656	< 83.6811
> 78.4031	> 82.1560	> 85.4775	> 86.2855
v 75.7421	v 79.6941	v 83.8802	v 88.2906
^ 75.1651	^ 78.6971	^ 82.9876	^ 86.0858
< 75.2733	< 74.8882	< 79.0958	< 85.2334
> 79.0846	> 82.7879	> 86.5027	> 89.2501
v 73.4680	v 75.1054	v 71.3707	v 91.3278
^ 76.1646	^ 77.7982	^ 64.5779	^ 87.4135
< 70.9652	< 69.6120	< 55.0120	< 73.2016
> 73.7186	> 67.1438	> 70.1177	> 92.6764
v 67.5342	v 65.8274	v 51.9917	v 94.4558
^ 70.4370	^ 70.2058	^ 0.0000	^ 100.0000
< 65.1671	< 61.8682	< 0.0000	< 100.0000
> 60.6361	> 64.1566	> 0.0000	> 100.0000
v 64.7431	v 62.9245	v 0.0000	v 100.0000

Figure 4.20: Q values for the $\varepsilon = 0.2$, $i = 5000000$.

On the basis of the above tests, it was observed that for both values of the ε parameter, the policy of moves corresponds to the policy of the value iteration algorithm after 100000 iterations. With the increasing number of iterations, the utility results come closer to the results achieved in the value iteration algorithm. For a larger ε, some trials show better results, which is associated with more frequent exploration. However, this conclusion is not consistent for 5000000 iterations. This may be due to the randomness of the solution itself. In addition, it was observed that better results are obtained near the terminal state of the world. The performance of the QLearning algorithm was also tested for $\varepsilon = 0.5$ and 5000000 iterations. The results of this trial are shown in fig. 4.21. Here, the best results were obtained compared to previous tests.

	1	2	3	4
4	> 81.6645	> 84.1246	> 86.5159	v 88.8446
3	> 81.4548	> 84.1583	> 87.0272	v 91.5431
2	^ 79.1076	^ 80.3618	B > 70.4658	v 94.5331
1	S ^ 76.5594	^ 77.6570	F 0.0000	T 100.0000

Figure 4.21: Utilities and policy for the $\varepsilon = 0.5$, $i = 5000000$.

5 MDP results for the additional test world

The additional test world has been created in the shape of a maze. There are two paths to the terminal state. One of them is longer because it requires 10 moves, and the other is shorter and requires only 8 moves. However, a special state with a large negative reward has been placed on the shorter path. Thus, the algorithm suggests taking a longer path. Figure 5.1 shows the result of the value iteartion algorithm for the created world. Algorithm converges in 25 iterations.

```
World Parameters:
Width X: 5
Height Y: 5
Start X: 5
Start Y: 5
Uncertainty Distribution P: 0.8(^), 0.1(<), 0.1(>)
Reward: -1
Discounting Parameter Gamma: 0.99
Exploration Parameter Epsilon: 0.25
(T) Terminal States: (1,1,100)
(B) Special States: (5,2,-50)
(F) Forbidden States: (1,2) (5,1) (3,2) (3,3) (4,3) (4,4) (2,4) (3,4)
```

	1	2	3	4	5
5	v 84.3240	< 82.0257	< 79.7561	< 77.5147	S < 75.0319
4	v 86.9386	F 0.0000	F 0.0000	F 0.0000	^ 72.8495
3	> 89.2990	v 91.9841	F 0.0000	F 0.0000	^ 70.6942
2	F 0.0000	v 94.7438	F 0.0000	v 81.6496	B < 24.0457
1	T 100.0000	< 97.2027	< 94.7438	< 91.1436	F 0.0000

Figure 5.1: Utilities and policy for the additional test world.

6 Q-learning results for the additional test world

	1	2	3	4	5
5	v 83.7089	< 80.7546	< 77.2524	< 72.8700	S < 66.1577
4	v 86.7337	F 0.0000	F 0.0000	F 0.0000	 ^ 58.8901
3	> 89.2099	 v 91.9601	F 0.0000	F 0.0000	 ^ 50.1402
2	F 0.0000	 v 94.7418	F 0.0000	 v 80.4329	B < 19.1729
1	T 100.0000	 < 97.2029	 < 94.7387	 < 90.9193	F 0.0000

Figure 6.1: Utilities and policy for the $\varepsilon = 0.5$, $i = 5000000$.

	1	2	3	4	5
5	^ 81.0878 < 81.7504 > 79.0770 v 83.7089	^ 78.0043 < 80.7546 > 74.7634 v 78.0007	^ 73.8143 < 77.2524 > 69.6759 v 73.8162	^ 68.3412 < 72.8700 > 62.1911 v 68.3374	^ 61.3937 < 66.1577 > 59.7418 v 54.6147
4	^ 82.0785 < 84.6244 > 84.6249 v 86.7337	^ 0.0000 < 0.0000 > 0.0000 v 0.0000	^ 0.0000 < 0.0000 > 0.0000 v 0.0000	^ 0.0000 < 0.0000 > 0.0000 v 0.0000	^ 58.8901 < 51.9701 > 51.9751 v 44.2732
3	^ 85.4836 < 86.9886 > 89.2099 v 87.5199	^ 89.7418 < 88.0730 > 90.2967 v 91.9601	^ 0.0000 < 0.0000 > 0.0000 v 0.0000	^ 0.0000 < 0.0000 > 0.0000 v 0.0000	^ 50.1402 < 40.6363 > 40.6308 v 21.3452
2	^ 0.0000 < 0.0000 > 0.0000 v 0.0000	^ 90.5740 < 92.7602 > 92.7597 v 94.7418	^ 0.0000 < 0.0000 > 0.0000 v 0.0000	^ 71.8441 < 79.0215 > 29.4157 v 80.4329	^ -5.6620 < 19.1729 > -30.3986 v -26.7271
1	^ 100.0000 < 100.0000 > 100.0000 v 100.0000	^ 93.3123 < 97.2029 > 93.0275 v 95.2598	^ 92.6327 < 94.7387 > 89.6298 v 92.6357	^ 80.5425 < 90.9193 > 87.7532 v 89.2394	^ 0.0000 < 0.0000 > 0.0000 v 0.0000

Figure 6.2: Q values for the $\varepsilon = 0.5$, $i = 5000000$.