Group Members: Talha Mohammed (UHID: 2031877), Dylan Berens (UHID: 1899838), Sai Bharani Veerepalli (UHID: 2209460), Muhammad Talha Abdullah (UHUD:2299629), Mahmoud Masoud (UHID: 2227225)

Task 4 Report

Experiment 1a Results:

Experiment 1a: PRANDOM for all 8000 steps
Running Experiment 1a (Seed: 42)
Terminal state 1 reached at step 740
Episode steps: 740, Episode reward: -180
Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1680
Episode steps: 940, Episode reward: -380
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2500
Episode steps: 820, Episode reward: -260
Step 3000/8000 - Terminals: 3

Terminal state 4 reached at step 3052
Episode steps: 552, Episode reward: 8
Step 4000/8000 - Terminals: 4

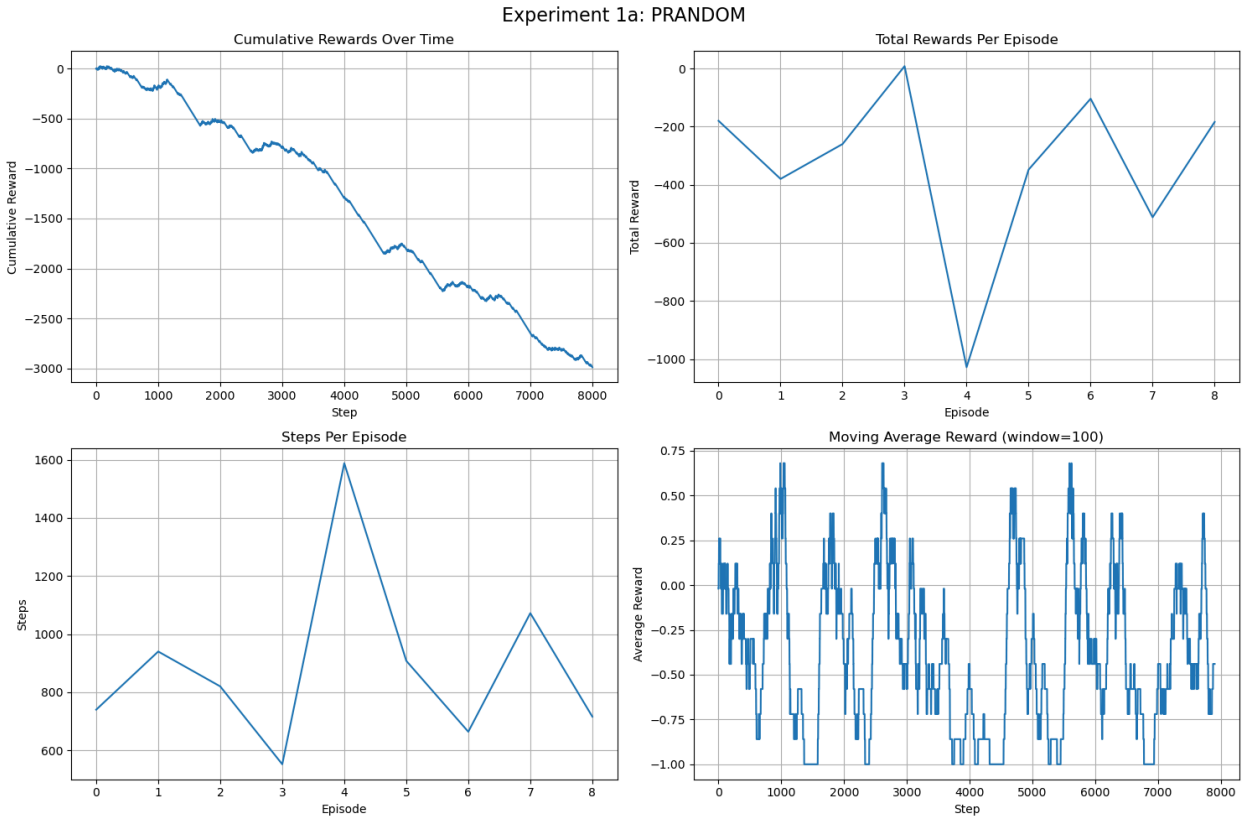
Terminal state 5 reached at step 4640
Episode steps: 1588, Episode reward: -1028
Step 5000/8000 - Terminals: 5

Terminal state 6 reached at step 5548
Episode steps: 908, Episode reward: -348
Step 6000/8000 - Terminals: 6

Terminal state 7 reached at step 6212
Episode steps: 664, Episode reward: -104
Step 7000/8000 - Terminals: 7

Terminal state 8 reached at step 7284
Episode steps: 1072, Episode reward: -512
Step 8000/8000 - Terminals: 8

Experiment completed:
Total terminal states reached: 8
Average steps per episode: 888.89
Average reward per episode: -332.00
Average agent distance per episode: 3.18



Experiment 1b Results:
Experiment 1b: PRANDOM (500 steps) then PGREEDY (7500 steps)
Running Experiment 1b (Seed: 42)

Switching to policy PGREEDY at step 500

Terminal state 1 reached at step 896
Episode steps: 896, Episode reward: -336
Step 1000/8000 - Terminals: 1

Step 2000/8000 - Terminals: 1

Terminal state 2 reached at step 2156
Episode steps: 1260, Episode reward: -700
Step 3000/8000 - Terminals: 2

Terminal state 3 reached at step 3068
Episode steps: 912, Episode reward: -352
Terminal state 4 reached at step 3856
Episode steps: 788, Episode reward: -228
Step 4000/8000 - Terminals: 4

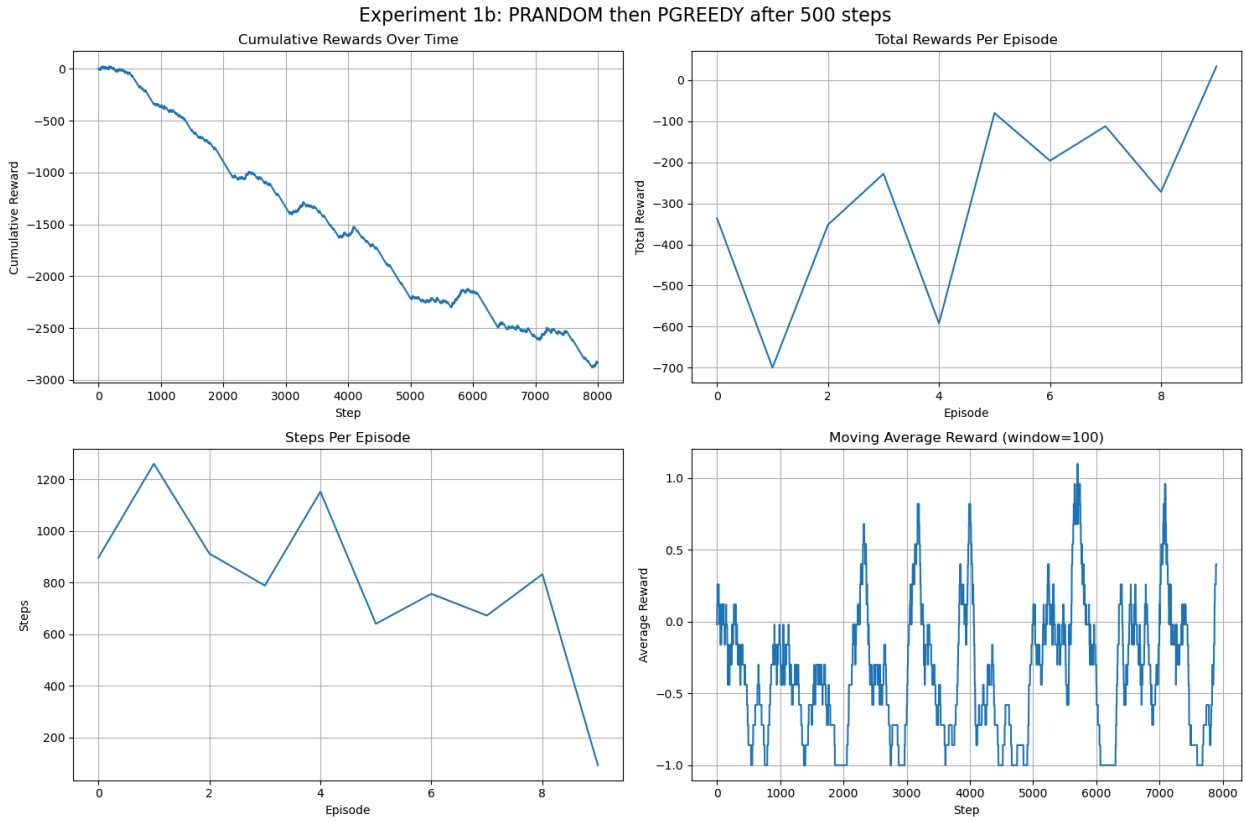
Step 5000/8000 - Terminals: 4

Terminal state 5 reached at step 5008
Episode steps: 1152, Episode reward: -592
Terminal state 6 reached at step 5648
Episode steps: 640, Episode reward: -80
Step 6000/8000 - Terminals: 6

Terminal state 7 reached at step 6404
Episode steps: 756, Episode reward: -196
Step 7000/8000 - Terminals: 7

Terminal state 8 reached at step 7076
Episode steps: 672, Episode reward: -112
Terminal state 9 reached at step 7908
Episode steps: 832, Episode reward: -272
Step 8000/8000 - Terminals: 9

Experiment completed:
Total terminal states reached: 9
Average steps per episode: 800.00
Average reward per episode: -283.40
Average agent distance per episode: 3.22



Experiment 1c Results:
Experiment 1c: PRANDOM (500 steps) then PEXPLOIT (7500 steps)

Running Experiment 1c (Seed: 42)

Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 796
Episode steps: 796, Episode reward: -236
Step 1000/8000 - Terminals: 1

Step 2000/8000 - Terminals: 1

Terminal state 2 reached at step 2104
Episode steps: 1308, Episode reward: -748
Step 3000/8000 - Terminals: 2

Terminal state 3 reached at step 3360
Episode steps: 1256, Episode reward: -696
Step 4000/8000 - Terminals: 3

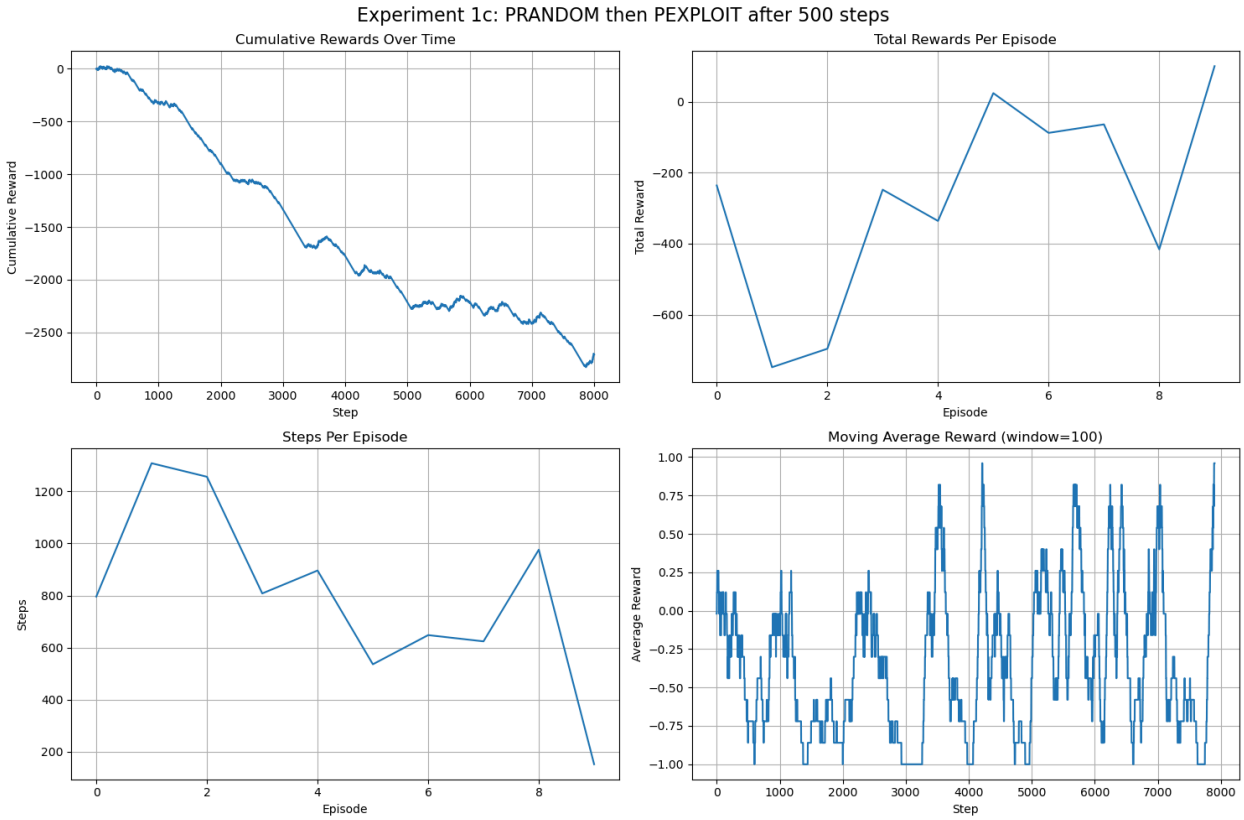
Terminal state 4 reached at step 4168
Episode steps: 808, Episode reward: -248
Step 5000/8000 - Terminals: 4

Terminal state 5 reached at step 5064
Episode steps: 896, Episode reward: -336
Terminal state 6 reached at step 5600
Episode steps: 536, Episode reward: 24
Step 6000/8000 - Terminals: 6

Terminal state 7 reached at step 6248
Episode steps: 648, Episode reward: -88
Terminal state 8 reached at step 6872
Episode steps: 624, Episode reward: -64
Step 7000/8000 - Terminals: 8

Terminal state 9 reached at step 7848
Episode steps: 976, Episode reward: -416
Step 8000/8000 - Terminals: 9

Experiment completed:
Total terminal states reached: 9
Average steps per episode: 800.00
Average reward per episode: -270.80
Average agent distance per episode: 3.22



Coordination between agents in Exp. 1b and 1c:

Experiment 1b - Mean Agent Distance per Episode: 3.22

Experiment 1c - Mean Agent Distance per Episode: 3.22

Expiremtnt 1c Final Q-Table:

Final Q-Table (F):

state	N	S	E	W
((1, 3), False, (5, 3))	0.00000	-0.93048	-0.996336	-0.96198
((1, 2), False, (5, 3))	0.00000	-0.51000	-0.640050	-0.51000
((1, 2), False, (4, 3))	0.00000	-0.30000	-0.376500	-0.58650
((2, 2), False, (4, 3))	-0.65700	-0.65700	-0.555000	-0.65700
((2, 2), False, (5, 3))	-0.30000	-0.30000	-0.555000	0.00000
((1, 2), False, (5, 2))	0.00000	-0.30000	-0.300000	-0.62475
((2, 2), False, (5, 2))	-0.55500	-0.51000	-0.345000	-0.51000
((2, 2), False, (4, 2))	-0.58650	-0.37650	-0.345000	-0.30000
((1, 2), False, (4, 2))	0.00000	-0.70200	-0.510000	-0.55500
((2, 1), False, (5, 2))	-0.38325	0.00000	-0.300000	0.00000
((2, 1), False, (5, 3))	-0.51000	-0.51000	-0.657000	0.00000
((3, 1), False, (5, 3))	0.00000	0.00000	-0.586500	0.00000

```
Final Q-Table (M):
                  state        N       S       E       W
((5, 3), False, (1, 2)) -0.51000  0.0000 -0.6570 -0.5100
((4, 3), False, (1, 2))  0.00000 -0.3450  0.0000 -0.3000
((4, 3), False, (2, 2)) -0.30000 -0.3000 -0.3000  0.0000
((5, 3), False, (2, 2)) -0.30000  0.0000 -0.3000 -0.3450
((5, 2), False, (1, 2)) -0.30000  0.0000  0.0000  0.0000
((5, 2), False, (2, 2)) -0.30000  0.0000 -0.3000 -0.5100
((4, 2), False, (2, 2)) -0.49515  0.0000 -0.3000 -0.3000
((4, 2), False, (1, 2)) -0.30000 -0.3000  0.0000  0.0000
 ((4, 2), True, (1, 2))  0.00000 -0.3765 -0.3000 -0.3450
 ((4, 2), True, (2, 2)) -0.75990 -0.6570 -0.7599 -0.7020
 ((5, 2), True, (2, 2)) -0.39855  0.0000 -0.3765 -0.5865
 ((5, 2), True, (2, 1)) -0.51000  0.0000 -0.6570 -0.5100
```

Experiment 2 Results:
Experiment 2: SARSA with PRANDOM (500 steps) then PEXPLOIT (7500 steps)
Running Experiment 2 (SARSA) (Seed: 42)

Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 796
Episode steps: 796, Episode reward: -236
Step 1000/8000 - Terminals: 1

Step 2000/8000 - Terminals: 1

Terminal state 2 reached at step 2104
Episode steps: 1308, Episode reward: -748
Step 3000/8000 - Terminals: 2

Terminal state 3 reached at step 3172
Episode steps: 1068, Episode reward: -508
Step 4000/8000 - Terminals: 3

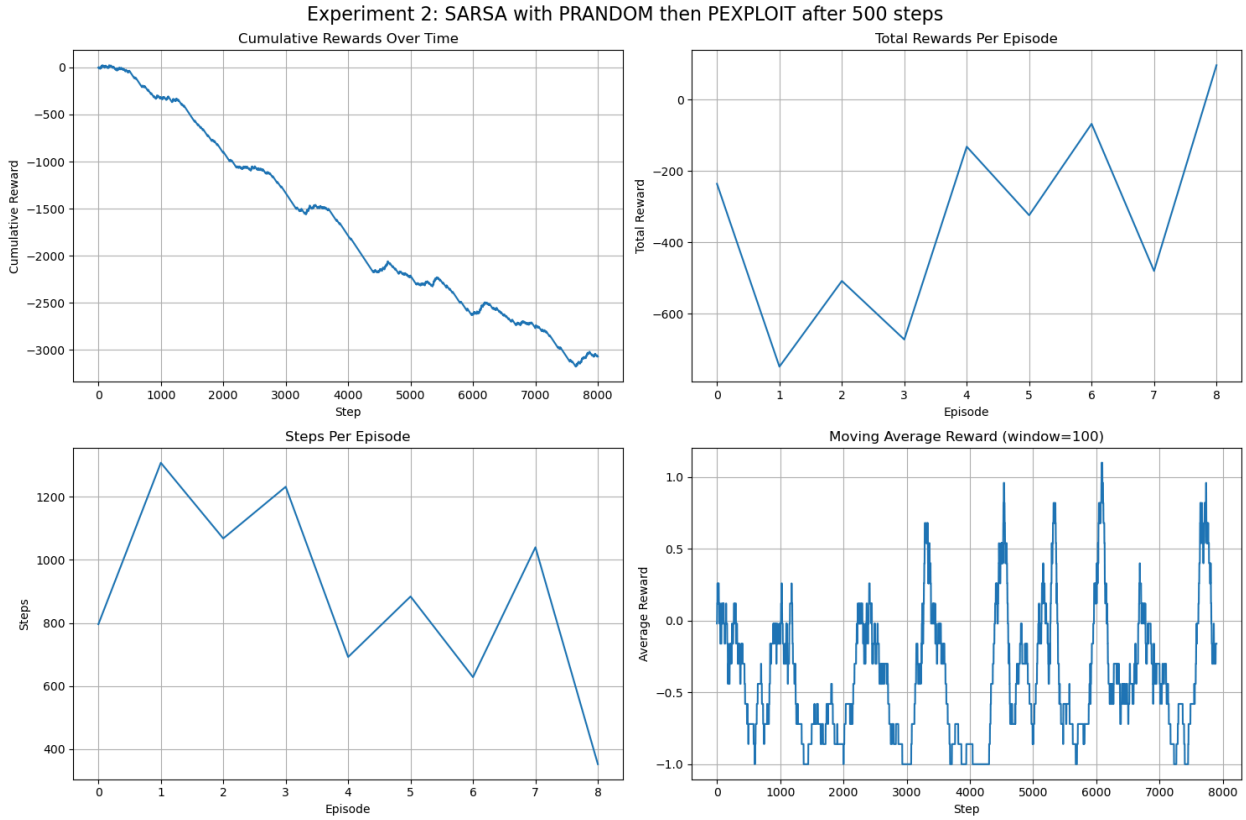
Terminal state 4 reached at step 4404
Episode steps: 1232, Episode reward: -672
Step 5000/8000 - Terminals: 4

Terminal state 5 reached at step 5096
Episode steps: 692, Episode reward: -132
Terminal state 6 reached at step 5980
Episode steps: 884, Episode reward: -324
Step 6000/8000 - Terminals: 6

Terminal state 7 reached at step 6608
Episode steps: 628, Episode reward: -68
Step 7000/8000 - Terminals: 7

Terminal state 8 reached at step 7648
Episode steps: 1040, Episode reward: -480
Step 8000/8000 - Terminals: 8

Experiment completed:
Total terminal states reached: 8
Average steps per episode: 888.89
Average reward per episode: -341.33
Average agent distance per episode: 3.23



Q-Table at first drop-off filled (F):

state	N	S	E	W
((1, 3), False, (5, 3))	0.0	0.000	0.0	-0.3
((1, 2), False, (5, 3))	0.0	0.000	0.0	0.0
((1, 2), False, (4, 3))	0.0	-0.300	0.0	-0.3
((2, 2), False, (4, 3))	0.0	0.000	0.0	0.0
((2, 2), False, (5, 3))	-0.3	0.000	0.0	0.0
((1, 2), False, (5, 2))	0.0	-0.300	0.0	0.0
((2, 2), False, (5, 2))	0.0	0.000	0.0	-0.3
((2, 2), False, (4, 2))	-0.3	0.000	0.0	0.0
((1, 2), False, (4, 2))	0.0	-0.345	-0.3	-0.3
((2, 1), False, (5, 2))	0.0	0.000	0.0	0.0
((2, 1), False, (5, 3))	0.0	-0.300	0.0	0.0
((3, 1), False, (5, 3))	0.0	0.000	0.0	0.0

Q-Table at first drop-off filled (M):

state	N	S	E	W
((5, 3), False, (1, 2))	-0.300	0.0	0.0	-0.3
((4, 3), False, (1, 2))	0.000	0.0	0.0	0.0
((4, 3), False, (2, 2))	0.000	-0.3	0.0	0.0
((5, 3), False, (2, 2))	-0.345	0.0	-0.3	0.0
((5, 2), False, (1, 2))	0.000	0.0	0.0	0.0
((5, 2), False, (2, 2))	-0.300	0.0	0.0	0.0
((4, 2), False, (2, 2))	0.000	0.0	0.0	0.0
((4, 2), False, (1, 2))	0.000	0.0	0.0	0.0
((4, 2), True, (1, 2))	0.000	0.0	0.0	0.0
((4, 2), True, (2, 2))	0.000	-0.3	0.0	0.0
((5, 2), True, (2, 2))	0.000	0.0	0.0	0.0
((5, 2), True, (2, 1))	0.000	0.0	-0.3	0.0

Q-Table at first terminal (F):

state	N	S	E	W
((1, 3), False, (5, 3))	0.0	0.000	0.0	-0.3
((1, 2), False, (5, 3))	0.0	0.000	0.0	0.0
((1, 2), False, (4, 3))	0.0	-0.300	0.0	-0.3
((2, 2), False, (4, 3))	0.0	0.000	-0.3	0.0
((2, 2), False, (5, 3))	-0.3	0.000	0.0	0.0
((1, 2), False, (5, 2))	0.0	-0.300	0.0	0.0
((2, 2), False, (5, 2))	0.0	0.000	0.0	-0.3
((2, 2), False, (4, 2))	-0.3	0.000	0.0	-0.3
((1, 2), False, (4, 2))	0.0	-0.345	-0.3	-0.3
((2, 1), False, (5, 2))	0.0	0.000	0.0	0.0
((2, 1), False, (5, 3))	-0.3	-0.510	0.0	0.0
((3, 1), False, (5, 3))	0.0	0.000	0.0	0.0

Q-Table at first terminal (M):

state	N	S	E	W
((5, 3), False, (1, 2))	-0.300	0.0	0.0	-0.3
((4, 3), False, (1, 2))	0.000	0.0	0.0	0.0
((4, 3), False, (2, 2))	-0.300	-0.3	0.0	0.0
((5, 3), False, (2, 2))	-0.345	0.0	-0.3	0.0
((5, 2), False, (1, 2))	0.000	0.0	0.0	0.0
((5, 2), False, (2, 2))	-0.300	0.0	0.0	0.0
((4, 2), False, (2, 2))	0.000	0.0	0.0	0.0
((4, 2), False, (1, 2))	0.000	0.0	0.0	0.0
((4, 2), True, (1, 2))	0.000	0.0	0.0	0.0
((4, 2), True, (2, 2))	0.000	-0.3	0.0	0.0
((5, 2), True, (2, 2))	0.000	0.0	0.0	0.0
((5, 2), True, (2, 1))	0.000	0.0	-0.3	0.0

Final Q-Table (F):

state	N	S	E	W
((1, 3), False, (5, 3))	0.000000	-0.804900	-0.80490	-1.006481
((1, 2), False, (5, 3))	0.000000	-0.300000	-0.64005	-0.555000
((1, 2), False, (4, 3))	0.000000	-0.300000	-0.62475	-0.300000
((2, 2), False, (4, 3))	-0.586500	-0.561750	-0.55500	-0.510000
((2, 2), False, (5, 3))	-0.300000	-0.300000	-0.30000	0.000000
((1, 2), False, (5, 2))	0.000000	-0.300000	-0.34500	-0.345000
((2, 2), False, (5, 2))	-0.555000	-0.510000	-0.58650	-0.561750
((2, 2), False, (4, 2))	-0.623985	-0.345000	-0.37650	-0.621189
((1, 2), False, (4, 2))	0.000000	-0.820335	-0.88368	-0.777184
((2, 1), False, (5, 2))	-0.345000	-0.345000	-0.30000	0.000000
((2, 1), False, (5, 3))	-0.555000	-0.510000	-0.51000	0.000000
((3, 1), False, (5, 3))	0.000000	0.000000	-0.34500	0.000000

Final Q-Table (M):

state	N	S	E	W
((5, 3), False, (1, 2))	-0.51000	0.00000	-0.75990	-0.510000
((4, 3), False, (1, 2))	0.00000	-0.34500	-0.34500	-0.300000
((4, 3), False, (2, 2))	-0.30000	-0.30000	-0.30000	0.285000
((5, 3), False, (2, 2))	-0.34500	0.00000	-0.30000	-0.490875
((5, 2), False, (1, 2))	-0.51000	0.00000	0.00000	0.000000
((5, 2), False, (2, 2))	0.07500	0.00000	-0.30000	-0.300000
((4, 2), False, (2, 2))	-0.25725	-0.28875	0.00000	-0.300000
((4, 2), False, (1, 2))	0.00000	0.00000	0.00000	0.000000
((4, 2), True, (1, 2))	0.00000	0.00000	-0.30000	0.000000
((4, 2), True, (2, 2))	-0.75990	-0.65700	-0.70200	-0.702000
((5, 2), True, (2, 2))	0.00000	0.00000	-0.34500	-0.300000
((5, 2), True, (2, 1))	-0.51000	0.00000	-0.56175	-0.510000

Experiment 3a Results:
Experiment 3a: Q-learning with alpha=0.15
Running Experiment 3a (alpha=0.15) (Seed: 42)

Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 796
Episode steps: 796, Episode reward: -236
Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1556
Episode steps: 760, Episode reward: -200
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2392
Episode steps: 836, Episode reward: -276
Step 3000/8000 - Terminals: 3

Terminal state 4 reached at step 3188
Episode steps: 796, Episode reward: -236
Step 4000/8000 - Terminals: 4

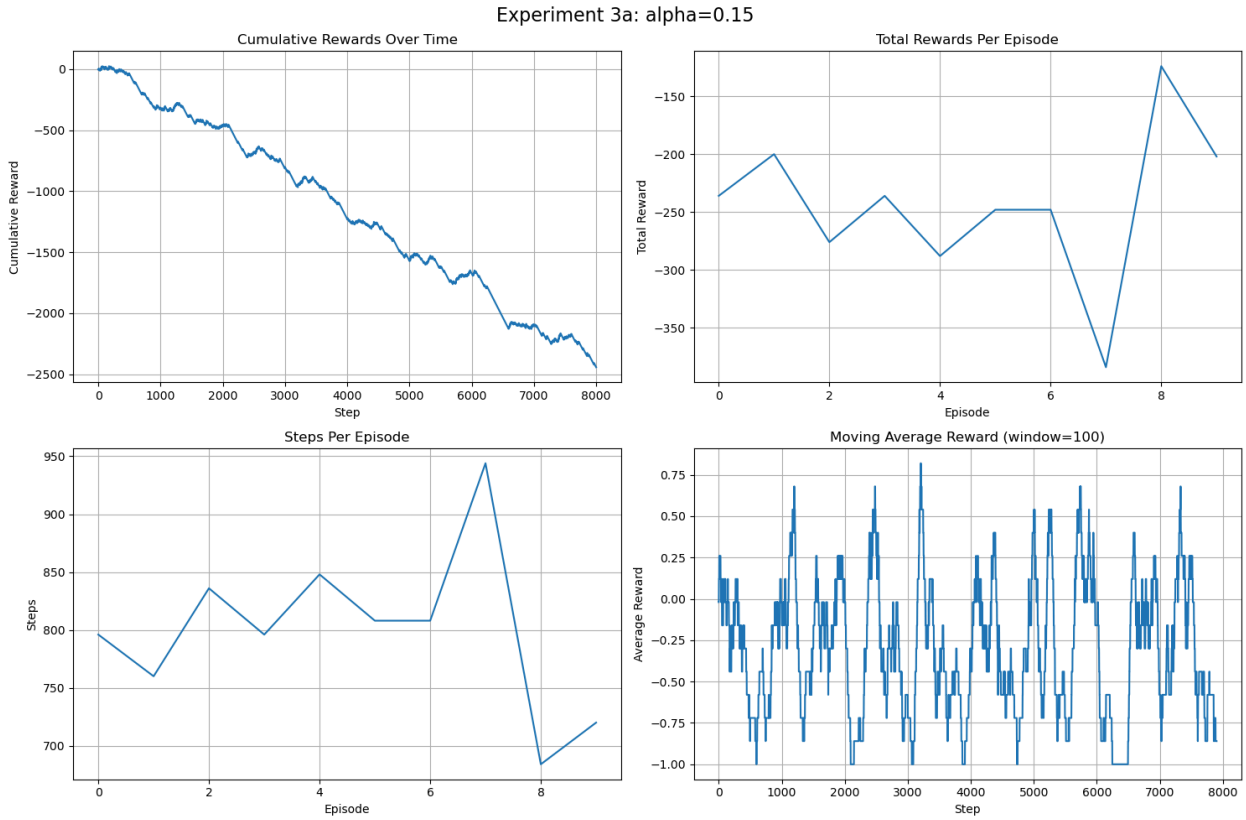
Terminal state 5 reached at step 4036
Episode steps: 848, Episode reward: -288
Terminal state 6 reached at step 4844
Episode steps: 808, Episode reward: -248
Step 5000/8000 - Terminals: 6

Terminal state 7 reached at step 5652
Episode steps: 808, Episode reward: -248
Step 6000/8000 - Terminals: 7

Terminal state 8 reached at step 6596
Episode steps: 944, Episode reward: -384
Step 7000/8000 - Terminals: 8

Terminal state 9 reached at step 7280
Episode steps: 684, Episode reward: -124
Step 8000/8000 - Terminals: 9

Experiment completed:
Total terminal states reached: 9
Average steps per episode: 800.00
Average reward per episode: -244.20
Average agent distance per episode: 3.27



Experiment 3b Results:
Experiment 3b: Q-learning with alpha=0.45
Running Experiment 3b (alpha=0.45) (Seed: 42)

Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 796
Episode steps: 796, Episode reward: -236
Step 1000/8000 - Terminals: 1

Step 2000/8000 - Terminals: 1

Terminal state 2 reached at step 2104
Episode steps: 1308, Episode reward: -748
Step 3000/8000 - Terminals: 2

Terminal state 3 reached at step 3356
Episode steps: 1252, Episode reward: -692
Step 4000/8000 - Terminals: 3

Terminal state 4 reached at step 4228
Episode steps: 872, Episode reward: -312
Terminal state 5 reached at step 4968
Episode steps: 740, Episode reward: -180

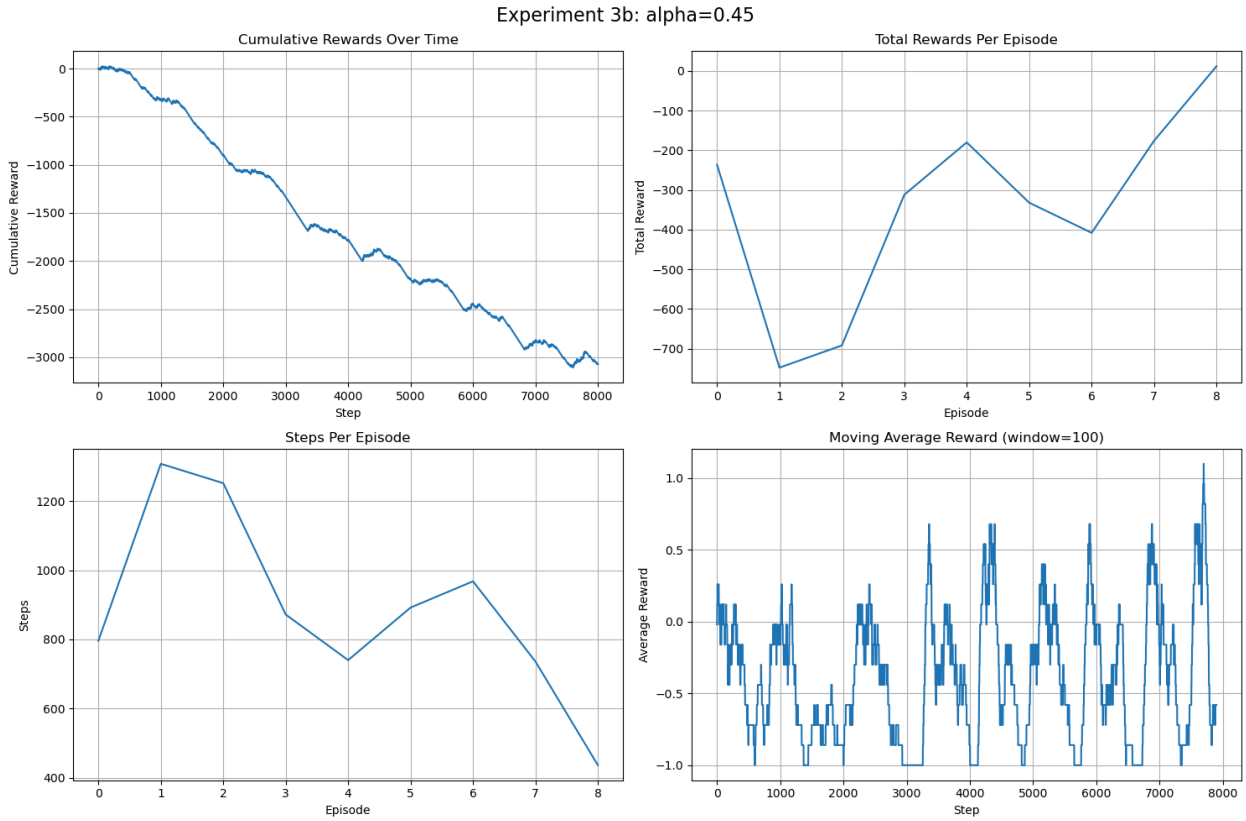
Step 5000/8000 - Terminals: 5

Terminal state 6 reached at step 5860
Episode steps: 892, Episode reward: -332
Step 6000/8000 - Terminals: 6

Terminal state 7 reached at step 6828
Episode steps: 968, Episode reward: -408
Step 7000/8000 - Terminals: 7

Terminal state 8 reached at step 7564
Episode steps: 736, Episode reward: -176
Step 8000/8000 - Terminals: 8

Experiment completed:
Total terminal states reached: 8
Average steps per episode: 888.89
Average reward per episode: -341.33
Average agent distance per episode: 3.19



Experiment 4 Results:
Experiment 4: Pickup locations change after 3rd terminal state
Running Experiment 4 (Seed: 42)

Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 796
Episode steps: 796, Episode reward: -236
Step 1000/8000 - Terminals: 1

Step 2000/8000 - Terminals: 1

Terminal state 2 reached at step 2104
Episode steps: 1308, Episode reward: -748
Step 3000/8000 - Terminals: 2

Terminal state 3 reached at step 3360
Episode steps: 1256, Episode reward: -696
Pickup locations changed to: [(1, 2), (4, 5)]
Terminal state 4 reached at step 3972
Episode steps: 612, Episode reward: -52
Step 4000/8000 - Terminals: 4

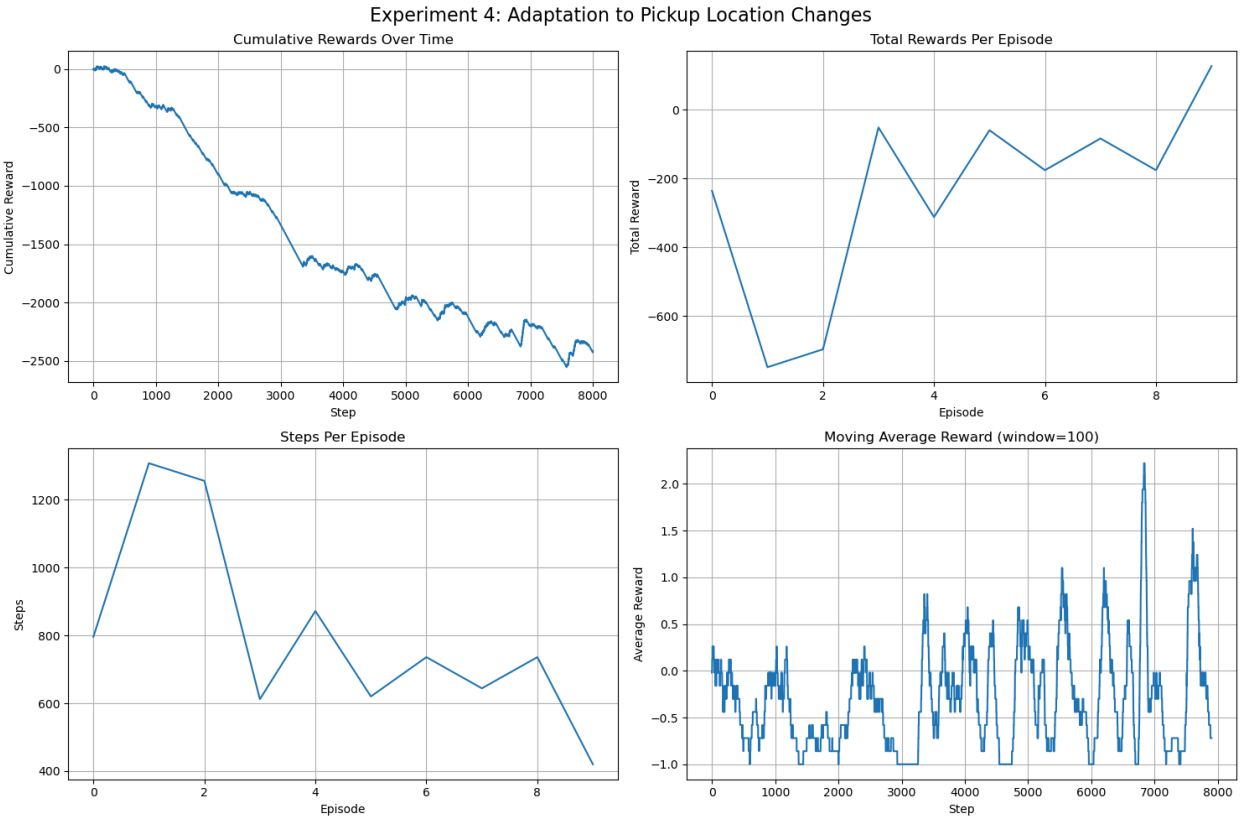
Terminal state 5 reached at step 4844
Episode steps: 872, Episode reward: -312
Step 5000/8000 - Terminals: 5

Terminal state 6 reached at step 5464
Episode steps: 620, Episode reward: -60
Step 6000/8000 - Terminals: 6

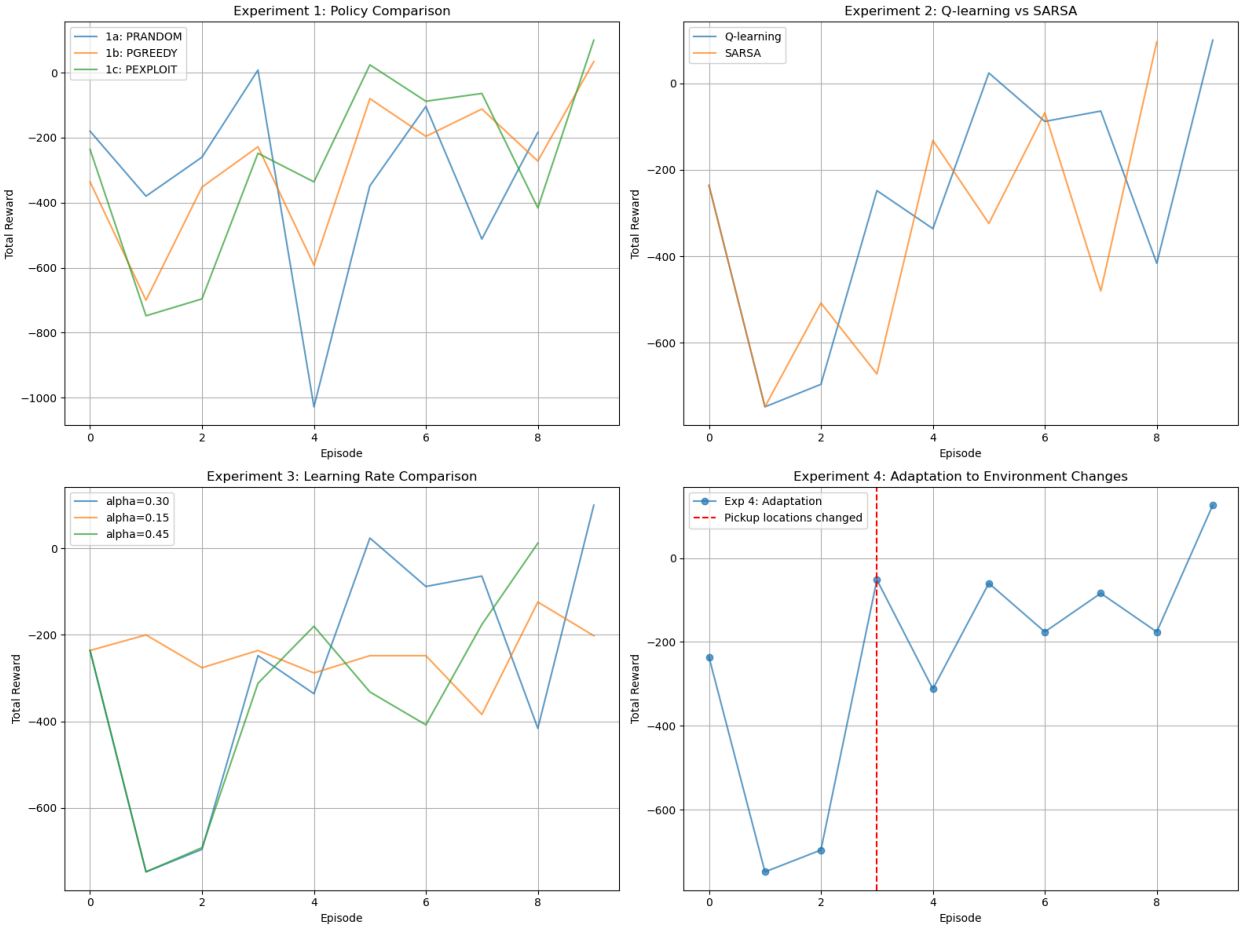
Terminal state 7 reached at step 6200
Episode steps: 736, Episode reward: -176
Terminal state 8 reached at step 6844
Episode steps: 644, Episode reward: -84
Step 7000/8000 - Terminals: 8

Terminal state 9 reached at step 7580
Episode steps: 736, Episode reward: -176
Step 8000/8000 - Terminals: 9

Experiment completed:
Total terminal states reached: 9
Average steps per episode: 800.00
Average reward per episode: -241.40
Average agent distance per episode: 3.39



Comparison of All Experiments(Run 1):



Summary Statistics:

1a (PRANDOM):
Terminal states reached: 8
Avg steps per episode: 888.89
Avg reward per episode: -332.00
Total cumulative reward: -2988.00

1b (PGREEDY):
Terminal states reached: 9
Avg steps per episode: 800.00
Avg reward per episode: -283.40
Total cumulative reward: -2834.00

1c (PEXPLOIT):
Terminal states reached: 9
Avg steps per episode: 800.00
Avg reward per episode: -270.80
Total cumulative reward: -2708.00

2 (SARSA):
Terminal states reached: 8

Avg steps per episode: 888.89
Avg reward per episode: -341.33
Total cumulative reward: -3072.00

3a (alpha=0.15):
Terminal states reached: 9
Avg steps per episode: 800.00
Avg reward per episode: -244.20
Total cumulative reward: -2442.00

3b (alpha=0.45):
Terminal states reached: 8
Avg steps per episode: 888.89
Avg reward per episode: -341.33
Total cumulative reward: -3072.00

4 (Adaptation):
Terminal states reached: 9
Avg steps per episode: 800.00
Avg reward per episode: -241.40
Total cumulative reward: -2414.00

Experiment 1a(Run 2) Results:
Running experiments again with different random seed for comparison
Running Experiment 1a Run 2 (Seed: 999)

Terminal state 1 reached at step 568
Episode steps: 568, Episode reward: -8
Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1512
Episode steps: 944, Episode reward: -384
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2428
Episode steps: 916, Episode reward: -356
Step 3000/8000 - Terminals: 3

Terminal state 4 reached at step 3236
Episode steps: 808, Episode reward: -248
Step 4000/8000 - Terminals: 4

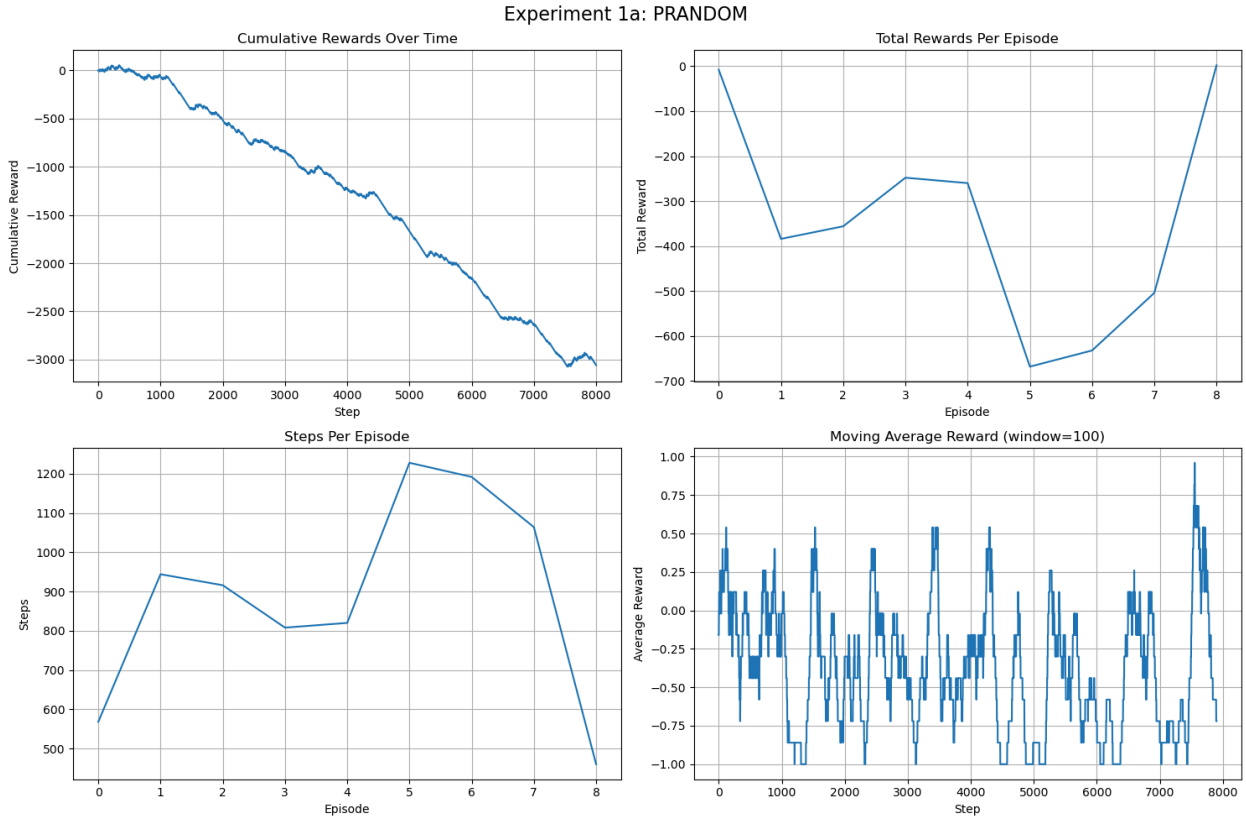
Terminal state 5 reached at step 4056
Episode steps: 820, Episode reward: -260
Step 5000/8000 - Terminals: 5

Terminal state 6 reached at step 5284
Episode steps: 1228, Episode reward: -668
Step 6000/8000 - Terminals: 6

Terminal state 7 reached at step 6476
Episode steps: 1192, Episode reward: -632
Step 7000/8000 - Terminals: 7

Terminal state 8 reached at step 7540
Episode steps: 1064, Episode reward: -504
Step 8000/8000 - Terminals: 8

Experiment completed:
Total terminal states reached: 8
Average steps per episode: 888.89
Average reward per episode: -339.78
Average agent distance per episode: 3.29



Experiment 1b(Run 2) Results:
Running Experiment 1b Run 2 (Seed: 999)
Switching to policy PGREEDY at step 500

Terminal state 1 reached at step 600
Episode steps: 600, Episode reward: -40

Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1296
Episode steps: 696, Episode reward: -136
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2284
Episode steps: 988, Episode reward: -428
Step 3000/8000 - Terminals: 3

Terminal state 4 reached at step 3380
Episode steps: 1096, Episode reward: -536
Step 4000/8000 - Terminals: 4

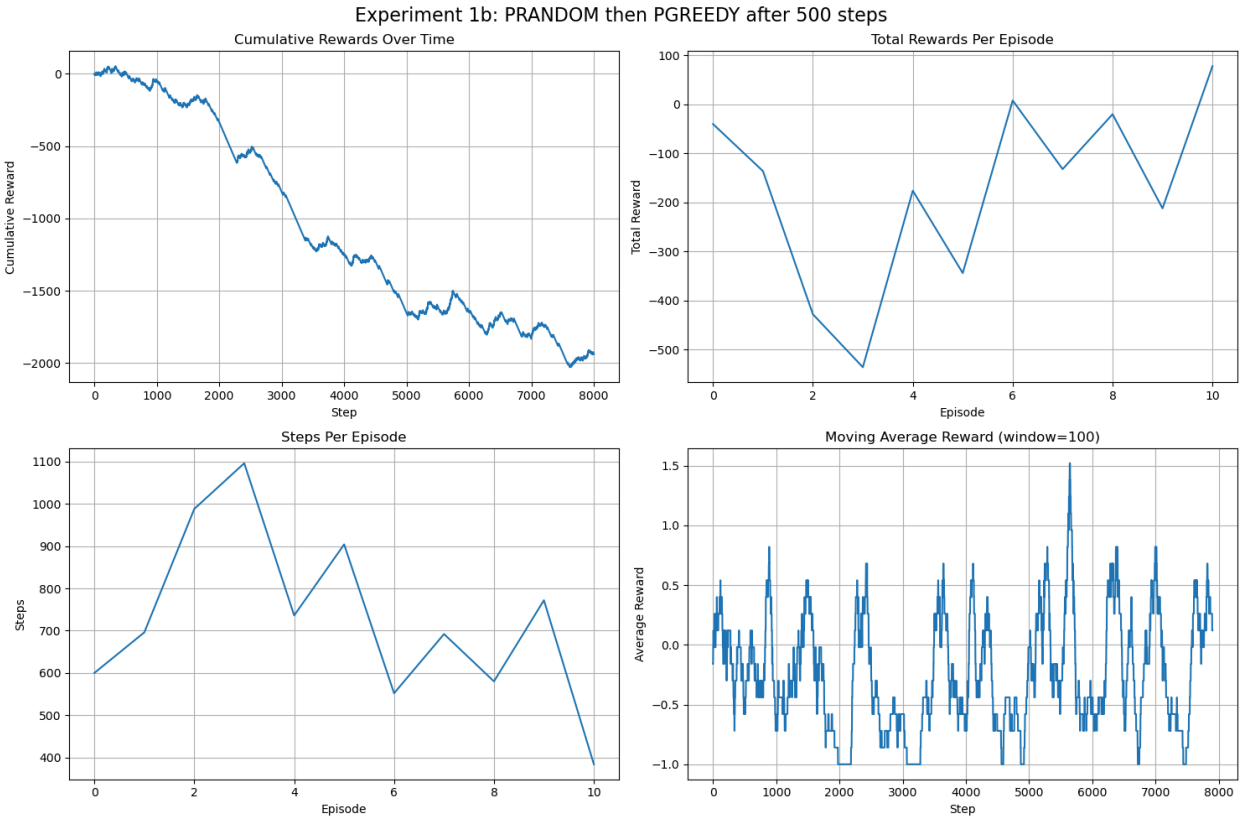
Terminal state 5 reached at step 4116
Episode steps: 736, Episode reward: -176
Step 5000/8000 - Terminals: 5

Terminal state 6 reached at step 5020
Episode steps: 904, Episode reward: -344
Terminal state 7 reached at step 5572
Episode steps: 552, Episode reward: 8
Step 6000/8000 - Terminals: 7

Terminal state 8 reached at step 6264
Episode steps: 692, Episode reward: -132
Terminal state 9 reached at step 6844
Episode steps: 580, Episode reward: -20
Step 7000/8000 - Terminals: 9

Terminal state 10 reached at step 7616
Episode steps: 772, Episode reward: -212
Step 8000/8000 - Terminals: 10

Experiment completed:
Total terminal states reached: 10
Average steps per episode: 727.27
Average reward per episode: -176.18
Average agent distance per episode: 3.18



Experiment 1c(Run 2) Results:
Running Experiment 1c Run 2 (Seed: 999)
Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 640
Episode steps: 640, Episode reward: -80
Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1472
Episode steps: 832, Episode reward: -272
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2492
Episode steps: 1020, Episode reward: -460
Step 3000/8000 - Terminals: 3

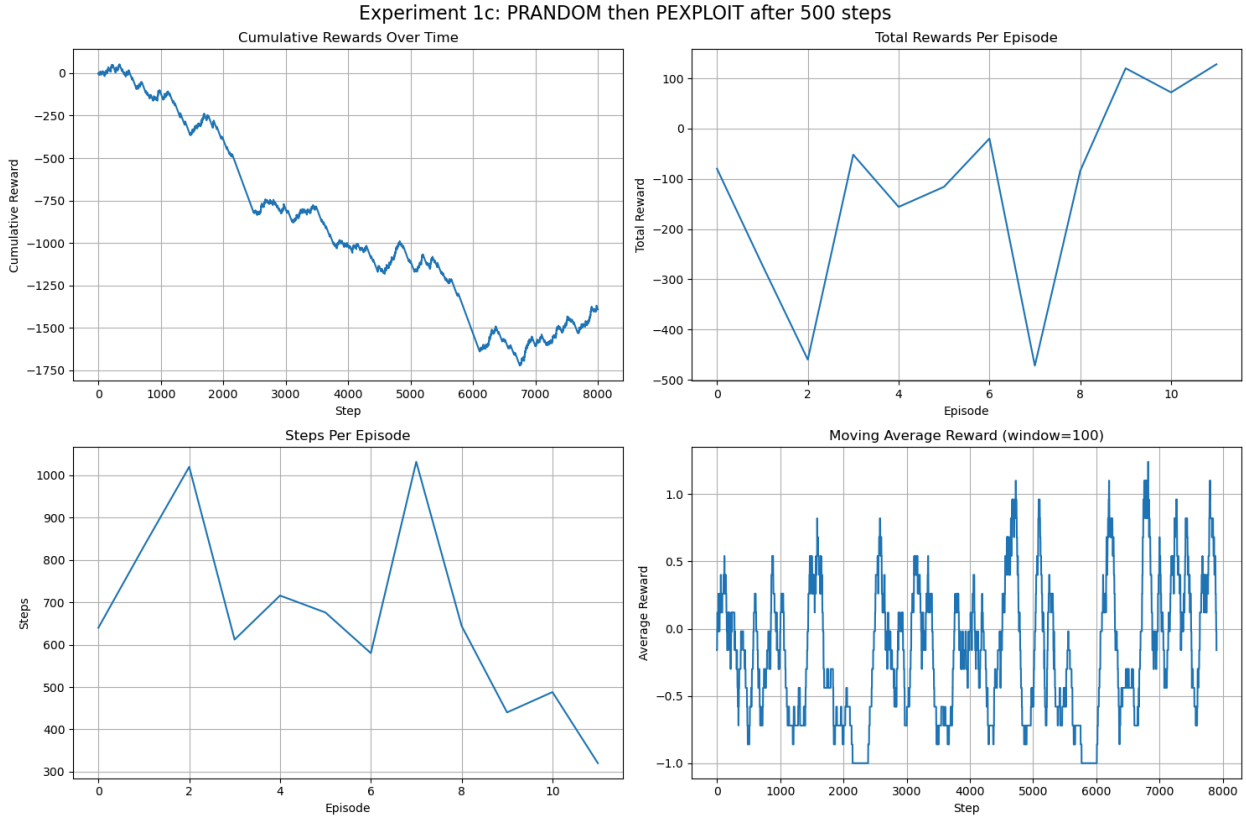
Terminal state 4 reached at step 3104
Episode steps: 612, Episode reward: -52
Terminal state 5 reached at step 3820
Episode steps: 716, Episode reward: -156
Step 4000/8000 - Terminals: 5
Terminal state 6 reached at step 4496
Episode steps: 676, Episode reward: -116
Step 5000/8000 - Terminals: 6

Terminal state 7 reached at step 5076
Episode steps: 580, Episode reward: -20
Step 6000/8000 - Terminals: 7

Terminal state 8 reached at step 6108
Episode steps: 1032, Episode reward: -472
Terminal state 9 reached at step 6752
Episode steps: 644, Episode reward: -84
Step 7000/8000 - Terminals: 9

Terminal state 10 reached at step 7192
Episode steps: 440, Episode reward: 120
Terminal state 11 reached at step 7680
Episode steps: 488, Episode reward: 72
Step 8000/8000 - Terminals: 11

Experiment completed:
Total terminal states reached: 11
Average steps per episode: 666.67
Average reward per episode: -116.00
Average agent distance per episode: 3.27



Coordination between agents in Exp. 1b and 1c:
Experiment 1b - Mean Agent Distance per Episode: 3.18

Experiment 1c - Mean Agent Distance per Episode: 3.27

Expiremtnt 1c Final Q-Table:

Final Q-Table (F):

state	N	S	E	W
((1, 3), False, (5, 3))	0.0	-0.83193	-0.882351	-0.882351
((1, 2), False, (5, 3))	0.0	0.00000	-0.413985	-0.755550
((1, 2), False, (5, 2))	0.0	0.00000	-0.300000	-0.510000
((1, 1), False, (5, 2))	0.0	-0.51000	-0.510000	0.000000
((1, 1), False, (5, 3))	0.0	-0.30000	-0.300000	0.000000
((1, 1), False, (4, 2))	0.0	-0.59325	-0.300000	0.000000
((1, 2), False, (4, 2))	0.0	-0.65700	-0.657000	-0.733500
((1, 1), False, (4, 1))	0.0	-0.65700	-0.657000	0.000000
((1, 2), False, (4, 1))	0.0	0.00000	-0.300000	-0.345000
((1, 2), False, (5, 1))	0.0	-0.51000	-0.510000	0.000000
((2, 2), False, (5, 1))	0.0	0.00000	0.000000	0.000000
((2, 2), False, (4, 1))	-0.3	-0.51000	-0.300000	-0.300000

Final Q-Table (M):

state	N	S	E	W
((5, 3), False, (1, 2))	-0.657	0.000	-0.51	-0.657
((5, 2), False, (1, 2))	0.000	0.000	0.00	0.000
((5, 2), False, (1, 1))	-0.300	0.000	-0.30	0.000
((5, 3), False, (1, 1))	0.000	0.000	-0.30	0.000
((4, 2), False, (1, 1))	0.000	0.000	0.00	0.000
((4, 2), False, (1, 2))	0.000	0.000	0.00	0.000
((4, 2), True, (1, 2))	0.000	0.000	-0.30	0.000
((4, 2), True, (1, 1))	-0.657	-0.510	-0.51	-0.657
((4, 1), True, (1, 1))	0.000	-0.300	-0.30	0.000
((4, 1), True, (1, 2))	-0.510	-0.555	-0.51	0.000
((5, 1), True, (1, 2))	-0.345	0.000	-0.30	0.000
((5, 1), True, (2, 2))	-0.555	0.000	-0.51	0.000

Experiment 2(Run 2) Results:
Running Experiment 2 Run 2 (Seed: 999)
Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 640
Episode steps: 640, Episode reward: -80
Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1472
Episode steps: 832, Episode reward: -272
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2420
Episode steps: 948, Episode reward: -388
Step 3000/8000 - Terminals: 3

Terminal state 4 reached at step 3512
Episode steps: 1092, Episode reward: -532
Step 4000/8000 - Terminals: 4

Terminal state 5 reached at step 4348
Episode steps: 836, Episode reward: -276
Terminal state 6 reached at step 4932
Episode steps: 584, Episode reward: -24
Step 5000/8000 - Terminals: 6

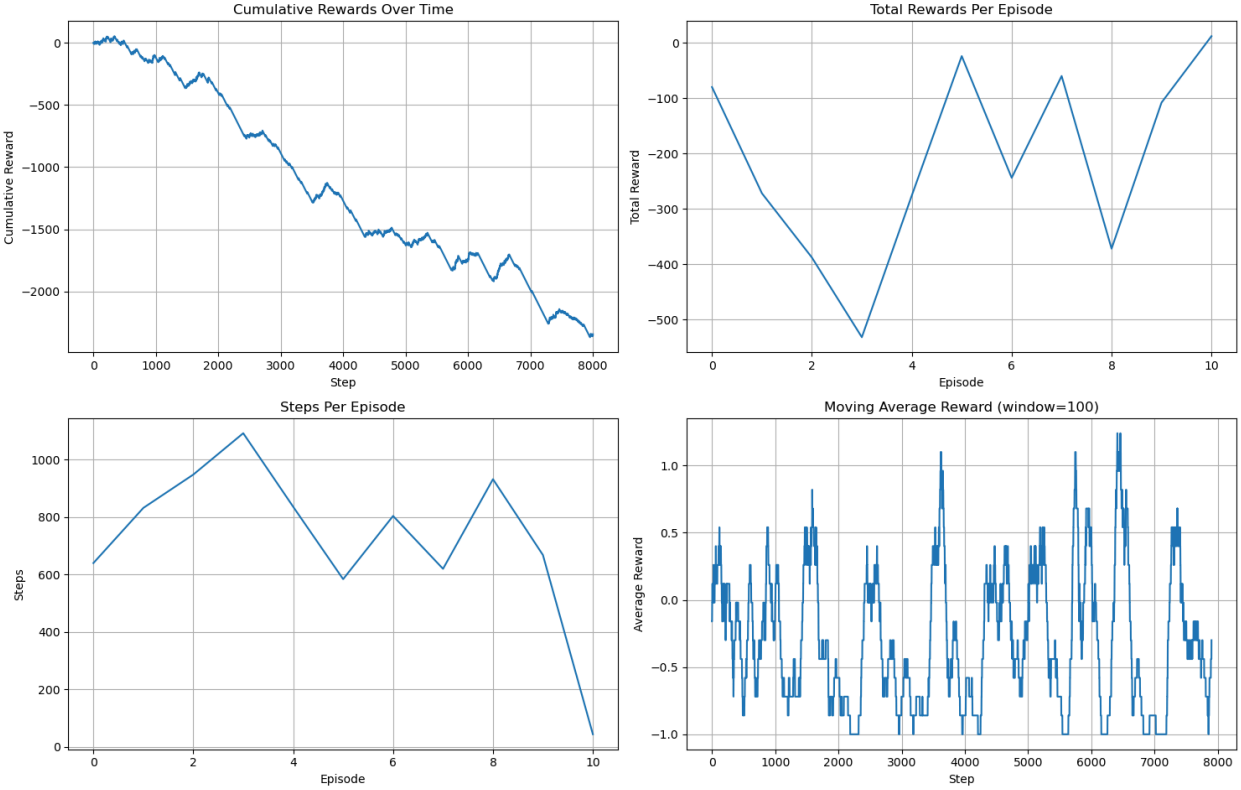
Terminal state 7 reached at step 5736
Episode steps: 804, Episode reward: -244
Step 6000/8000 - Terminals: 7

Terminal state 8 reached at step 6356
Episode steps: 620, Episode reward: -60
Step 7000/8000 - Terminals: 8

Terminal state 9 reached at step 7288
Episode steps: 932, Episode reward: -372
Terminal state 10 reached at step 7956
Episode steps: 668, Episode reward: -108
Step 8000/8000 - Terminals: 10

Experiment completed:
Total terminal states reached: 10
Average steps per episode: 727.27
Average reward per episode: -213.09
Average agent distance per episode: 3.29

Experiment 2: SARSA



Q-Table at first drop-off filled (F):

state	N	S	E	W
((1, 3), False, (5, 3))	0.0	0.0	0.0	-0.300
((1, 2), False, (5, 3))	0.0	0.0	0.0	0.000
((1, 2), False, (5, 2))	0.0	0.0	0.0	-0.510
((1, 1), False, (5, 2))	0.0	0.0	0.0	0.000
((1, 1), False, (5, 3))	0.0	0.0	-0.3	0.000
((1, 1), False, (4, 2))	0.0	0.0	-0.3	0.000
((1, 2), False, (4, 2))	0.0	0.0	0.0	-0.345
((1, 1), False, (4, 1))	0.0	0.0	-0.3	0.000
((1, 2), False, (4, 1))	0.0	0.0	0.0	0.000
((1, 2), False, (5, 1))	0.0	-0.3	0.0	0.000
((2, 2), False, (5, 1))	0.0	0.0	0.0	0.000
((2, 2), False, (4, 1))	0.0	0.0	-0.3	0.000

Q-Table at first drop-off filled (M):

state	N	S	E	W
((5, 3), False, (1, 2))	0.0	0.0	0.0	-0.51
((5, 2), False, (1, 2))	0.0	0.0	0.0	0.00
((5, 2), False, (1, 1))	-0.3	0.0	-0.3	0.00
((5, 3), False, (1, 1))	0.0	0.0	0.0	0.00
((4, 2), False, (1, 1))	0.0	0.0	0.0	0.00
((4, 2), False, (1, 2))	0.0	0.0	0.0	0.00
((4, 2), True, (1, 2))	0.0	0.0	0.0	0.00
((4, 2), True, (1, 1))	0.0	0.0	0.0	-0.30
((4, 1), True, (1, 1))	0.0	0.0	0.0	0.00
((4, 1), True, (1, 2))	0.0	-0.3	0.0	0.00
((5, 1), True, (1, 2))	0.0	0.0	0.0	0.00
((5, 1), True, (2, 2))	-0.3	0.0	0.0	0.00

Q-Table at first terminal (F):

state	N	S	E	W
((1, 3), False, (5, 3))	0.0	0.0	0.0	-0.300
((1, 2), False, (5, 3))	0.0	0.0	0.0	0.000
((1, 2), False, (5, 2))	0.0	0.0	0.0	-0.510
((1, 1), False, (5, 2))	0.0	0.0	0.0	0.000
((1, 1), False, (5, 3))	0.0	0.0	-0.3	0.000
((1, 1), False, (4, 2))	0.0	0.0	-0.3	0.000
((1, 2), False, (4, 2))	0.0	0.0	0.0	-0.345
((1, 1), False, (4, 1))	0.0	-0.3	-0.3	0.000
((1, 2), False, (4, 1))	0.0	0.0	0.0	0.000
((1, 2), False, (5, 1))	0.0	-0.3	0.0	0.000
((2, 2), False, (5, 1))	0.0	0.0	0.0	0.000
((2, 2), False, (4, 1))	0.0	0.0	-0.3	-0.300

Q-Table at first terminal (M):

state	N	S	E	W
((5, 3), False, (1, 2))	0.0	0.0	0.0	-0.51
((5, 2), False, (1, 2))	0.0	0.0	0.0	0.00
((5, 2), False, (1, 1))	-0.3	0.0	-0.3	0.00
((5, 3), False, (1, 1))	0.0	0.0	0.0	0.00
((4, 2), False, (1, 1))	0.0	0.0	0.0	0.00
((4, 2), False, (1, 2))	0.0	0.0	0.0	0.00
((4, 2), True, (1, 2))	0.0	0.0	0.0	0.00
((4, 2), True, (1, 1))	0.0	0.0	0.0	-0.30
((4, 1), True, (1, 1))	0.0	0.0	0.0	0.00
((4, 1), True, (1, 2))	0.0	-0.3	0.0	0.00
((5, 1), True, (1, 2))	0.0	0.0	0.0	0.00
((5, 1), True, (2, 2))	-0.3	0.0	-0.3	0.00

```
Final Q-Table (F):
                   state    N        S         E         W
((1, 3), False, (5, 3))  0.0 -0.87693 -0.949987 -0.958851
((1, 2), False, (5, 3))  0.0 -0.30000 -0.376500 -0.586500
((1, 2), False, (5, 2))  0.0  0.00000  0.000000 -0.510000
((1, 1), False, (5, 2))  0.0 -0.51000 -0.657000  0.000000
((1, 1), False, (5, 3))  0.0 -0.55500 -0.300000  0.000000
((1, 1), False, (4, 2))  0.0 -0.60855 -0.300000  0.000000
((1, 2), False, (4, 2))  0.0 -0.70200 -0.657000 -0.755550
((1, 1), False, (4, 1))  0.0 -0.51000 -0.657000  0.000000
((1, 2), False, (4, 1))  0.0  0.00000  0.000000 -0.345000
((1, 2), False, (5, 1))  0.0 -0.51000 -0.300000  0.000000
((2, 2), False, (5, 1))  0.0  0.00000  0.000000  0.000000
((2, 2), False, (4, 1)) -0.3 -0.65700 -0.300000 -0.510000
```

```
Final Q-Table (M):
                   state      N     S      E     W
((5, 3), False, (1, 2)) -0.657  0.00 -0.657 -0.51
((5, 2), False, (1, 2))  0.000  0.00 -0.618 -0.30
((5, 2), False, (1, 1)) -0.300  0.00 -0.300  0.00
((5, 3), False, (1, 1)) -0.300  0.00 -0.300  0.00
((4, 2), False, (1, 1)) -0.345  0.00  0.000  0.00
((4, 2), False, (1, 2))  0.000  0.00  0.000  0.00
 ((4, 2), True, (1, 2))  0.000  0.00 -0.300 -0.30
 ((4, 2), True, (1, 1)) -0.510 -0.51 -0.300 -0.51
 ((4, 1), True, (1, 1)) -0.345  0.00  0.000  0.00
 ((4, 1), True, (1, 2)) -0.555 -0.30 -0.510  0.00
 ((5, 1), True, (1, 2)) -0.300  0.00  0.000  0.00
 ((5, 1), True, (2, 2)) -0.555  0.00 -0.657  0.00
```

Experiment 3a(Run 2) Results:
Running Experiment 3a Run 2 (Seed: 999)
Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 640
Episode steps: 640, Episode reward: -80
Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1624
Episode steps: 984, Episode reward: -424
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2380
Episode steps: 756, Episode reward: -196
Step 3000/8000 - Terminals: 3

Terminal state 4 reached at step 3384
Episode steps: 1004, Episode reward: -444
Step 4000/8000 - Terminals: 4

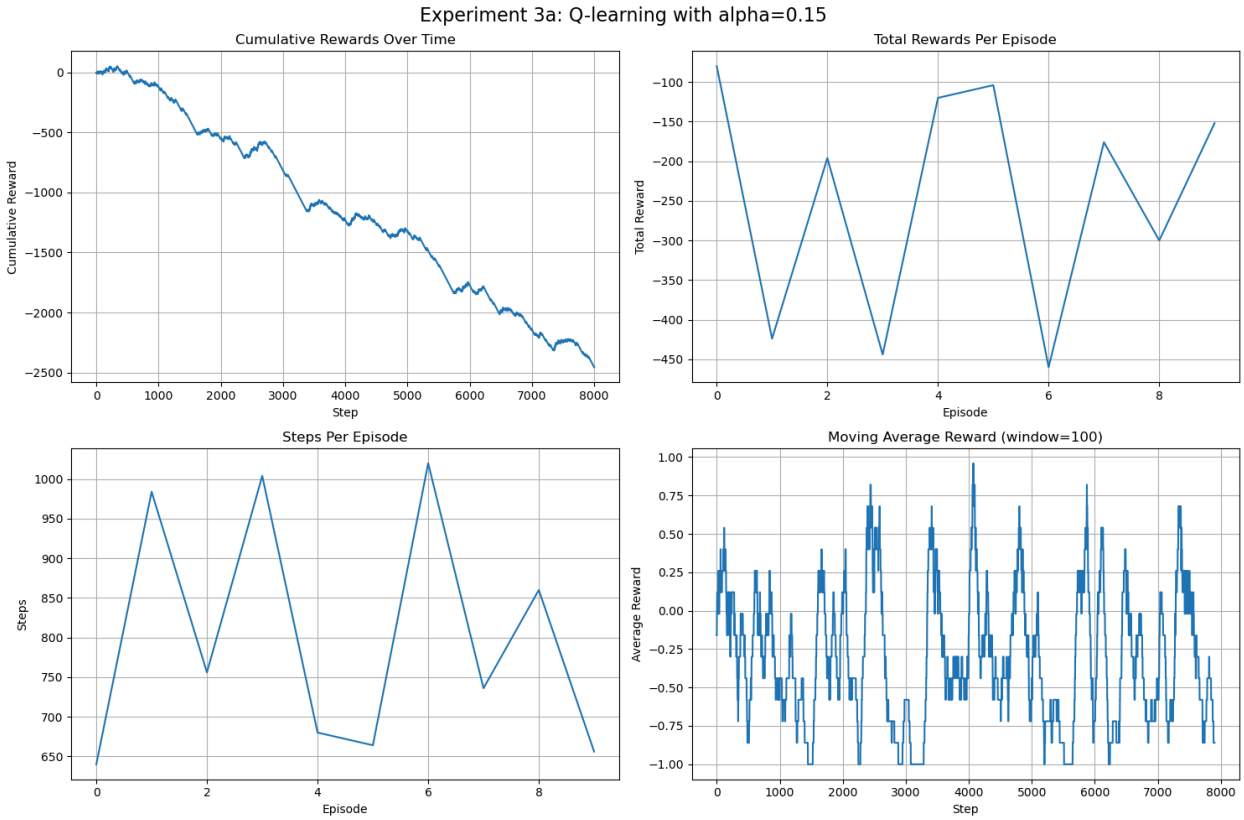
Terminal state 5 reached at step 4064
Episode steps: 680, Episode reward: -120
Terminal state 6 reached at step 4728
Episode steps: 664, Episode reward: -104
Step 5000/8000 - Terminals: 6

Terminal state 7 reached at step 5748
Episode steps: 1020, Episode reward: -460
Step 6000/8000 - Terminals: 7

Terminal state 8 reached at step 6484
Episode steps: 736, Episode reward: -176
Step 7000/8000 - Terminals: 8

Terminal state 9 reached at step 7344
Episode steps: 860, Episode reward: -300
Step 8000/8000 - Terminals: 9

Experiment completed:
Total terminal states reached: 9
Average steps per episode: 800.00
Average reward per episode: -245.60
Average agent distance per episode: 3.27



Experiment 3b(Run 2) Results:

Running Experiment 3b Run 2 (Seed: 999)
Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 640
Episode steps: 640, Episode reward: -80
Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1472
Episode steps: 832, Episode reward: -272
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2492
Episode steps: 1020, Episode reward: -460
Step 3000/8000 - Terminals: 3

Terminal state 4 reached at step 3228
Episode steps: 736, Episode reward: -176
Terminal state 5 reached at step 3996
Episode steps: 768, Episode reward: -208
Step 4000/8000 - Terminals: 5

Terminal state 6 reached at step 4788
Episode steps: 792, Episode reward: -232

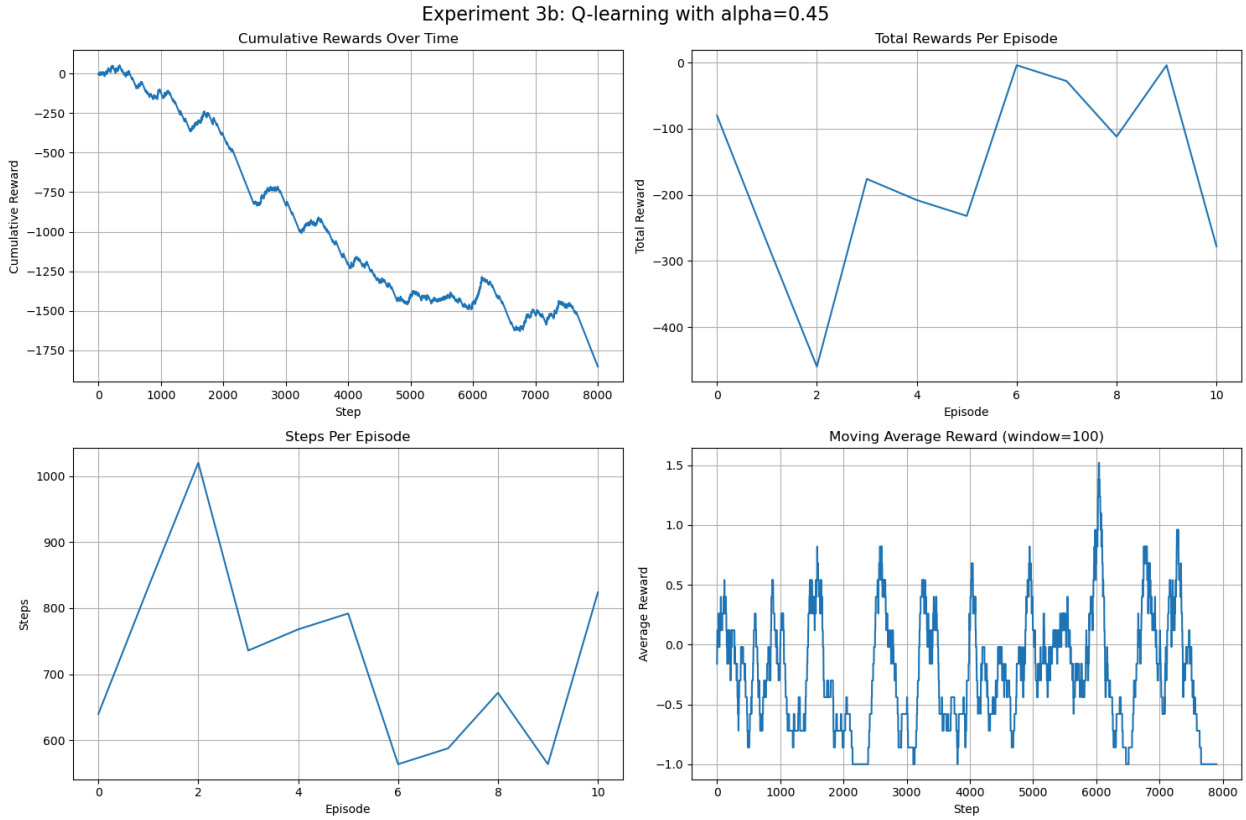
Step 5000/8000 - Terminals: 6

Terminal state 7 reached at step 5352
Episode steps: 564, Episode reward: -4
Terminal state 8 reached at step 5940
Episode steps: 588, Episode reward: -28
Step 6000/8000 - Terminals: 8

Terminal state 9 reached at step 6612
Episode steps: 672, Episode reward: -112
Step 7000/8000 - Terminals: 9

Terminal state 10 reached at step 7176
Episode steps: 564, Episode reward: -4
Step 8000/8000 - Terminals: 10

Experiment completed:
Total terminal states reached: 10
Average steps per episode: 727.27
Average reward per episode: -168.55
Average agent distance per episode: 3.26



Experiment 4(Run 2) Results:
Running Experiment 4 Run 2 (Seed: 999)
Switching to policy PEXPLOIT at step 500

Terminal state 1 reached at step 640
Episode steps: 640, Episode reward: -80
Step 1000/8000 - Terminals: 1

Terminal state 2 reached at step 1472
Episode steps: 832, Episode reward: -272
Step 2000/8000 - Terminals: 2

Terminal state 3 reached at step 2492
Episode steps: 1020, Episode reward: -460
Pickup locations changed to: [(1, 2), (4, 5)]
Step 3000/8000 - Terminals: 3

Step 4000/8000 - Terminals: 3

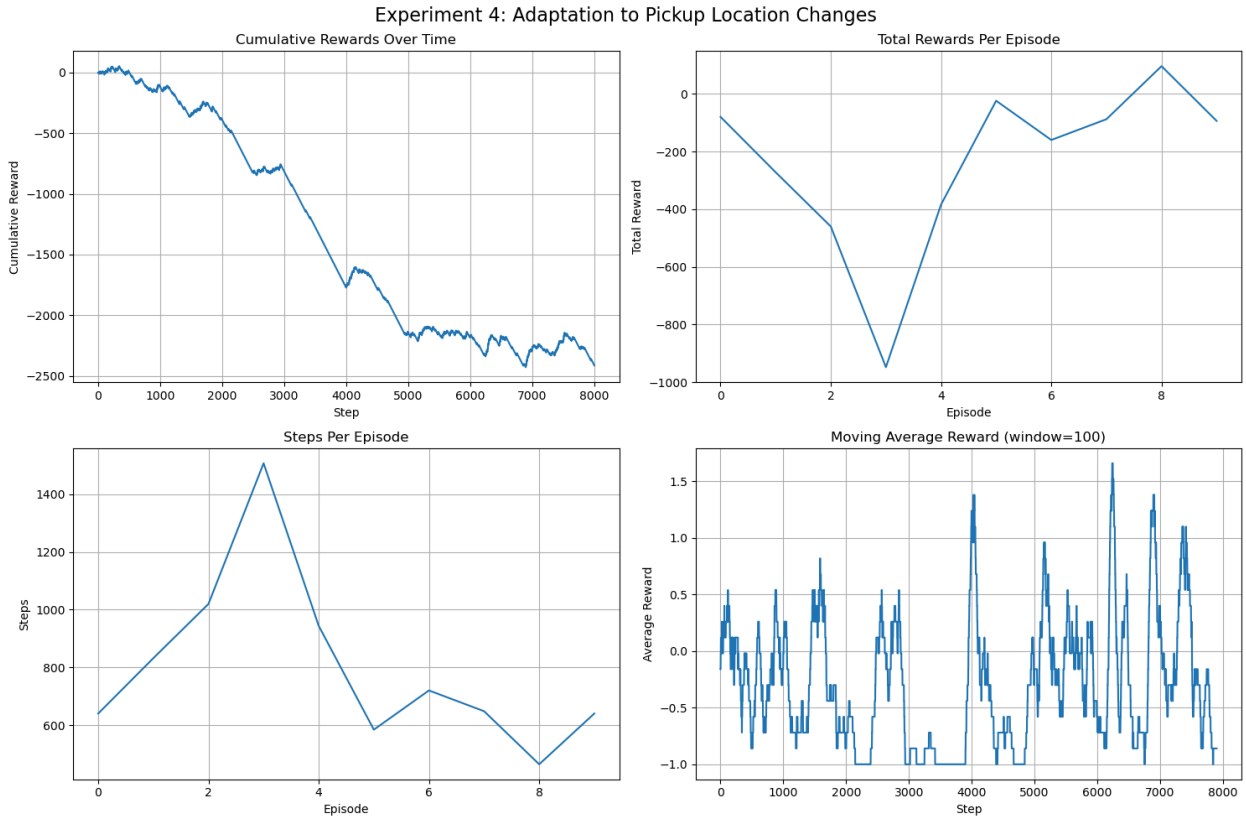
Terminal state 4 reached at step 4000
Episode steps: 1508, Episode reward: -948
Terminal state 5 reached at step 4944
Episode steps: 944, Episode reward: -384
Step 5000/8000 - Terminals: 5

Terminal state 6 reached at step 5528
Episode steps: 584, Episode reward: -24
Step 6000/8000 - Terminals: 6

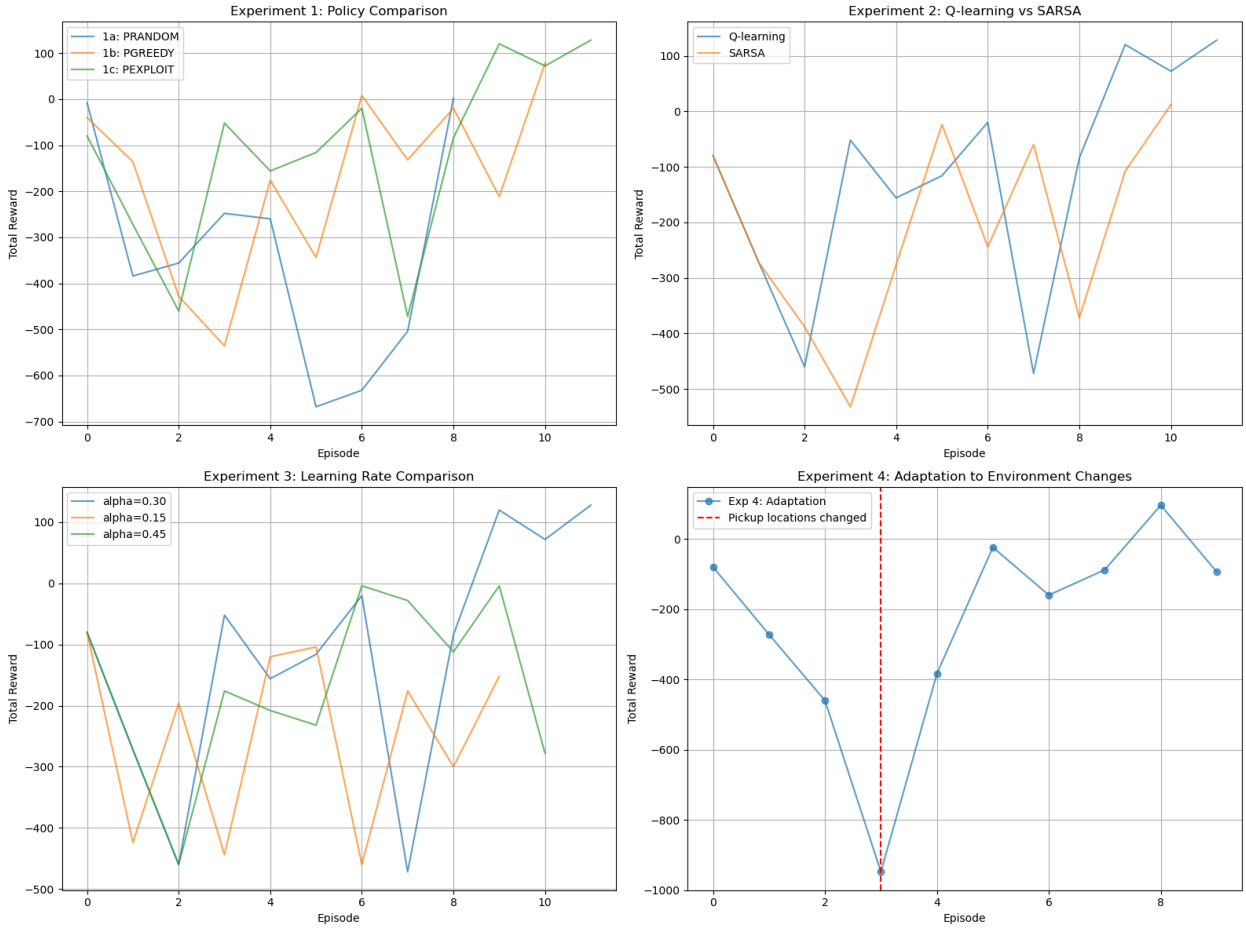
Terminal state 7 reached at step 6248
Episode steps: 720, Episode reward: -160
Terminal state 8 reached at step 6896
Episode steps: 648, Episode reward: -88
Step 7000/8000 - Terminals: 8

Terminal state 9 reached at step 7360
Episode steps: 464, Episode reward: 96
Step 8000/8000 - Terminals: 9

Experiment completed:
Total terminal states reached: 9
Average steps per episode: 800.00
Average reward per episode: -241.40
Average agent distance per episode: 3.31



Comparison of All Experiments(Run 2)



Summary Statistics

1a (PRANDOM):
Terminal states reached: 8
Avg steps per episode: 888.89
Avg reward per episode: -339.78
Total cumulative reward: -3058.00

1b (PGREEDY):
Terminal states reached: 10
Avg steps per episode: 727.27
Avg reward per episode: -176.18
Total cumulative reward: -1938.00

1c (PEXPLOIT):
Terminal states reached: 11
Avg steps per episode: 666.67
Avg reward per episode: -116.00
Total cumulative reward: -1392.00

2 (SARSA):

Terminal states reached: 10
Avg steps per episode: 727.27
Avg reward per episode: -213.09
Total cumulative reward: -2344.00

3a (alpha=0.15):
Terminal states reached: 9
Avg steps per episode: 800.00
Avg reward per episode: -245.60
Total cumulative reward: -2456.00

3b (alpha=0.45):
Terminal states reached: 10
Avg steps per episode: 727.27
Avg reward per episode: -168.55
Total cumulative reward: -1854.00

4 (Adaptation):
Terminal states reached: 9
Avg steps per episode: 800.00
Avg reward per episode: -241.40
Total cumulative reward: -2414.00

One of the key distinctions between our initial approach and our final code is that we decided to not have the agents be aware of the number of blocks at the pickup and dropoff locations. Accordingly, we adjusted the get_state function return to be only returning agent_pos, agent_holding, other_agnet_pos, and thereby makes the agent “state blind” to the number of blocks, significantly decreasing the size of the q table. We also altered our approach to action priority, with our final code making a hard coded priority rule that if pickup or dropoff is applicable, to always choose that action. PRANDOM, PGREEDY, PEXPLOIT policies are only applied to movement actions, exploring N/S/E/W. Additionally, we simplified our reward function to be the standard -1 constant pain for the agent to explore the space with any movement action, and +13 for a successful pickup or dropoff, with no more penalties directly tied to invalid moves as a result of the get_valid_actions function preventing them by limiting the agent to actions that are valid.

Our final code consists primarily of 4 sections: class PDWorld, class QLearningAgent, def run_experiment, and the helper functions including def plot_results for visualizing the results. The class PDWorld consists of 6 main functions after initialization. Def reset sets the world back to its initial state. Def is_terminal determines if the world is at its terminal state, wherein all drop off locations have 5 blocks. Def get_state simply returns the current state for a given agent. Def valid_actions is the logic for determining what possible actions can be taken in a current state. For example, if the agent is at the boundary of the space and cannot explore vertically, north

would not be a valid action. Similarly, if the agent is holding a block, then pickup will not be a valid option, nor will dropoff when holding nothing. This is an important function for pruning the search space by allowing the agent to only consider the actions that are actually possible. Def execute_action is the actual calling for a particular action to be taken. Def visualize allows the visualization of the current state of the world. Class QLearningAgent has the logic for the brain of this process, implementing the Q-learning/SARSA agent and the decision-making that stems from its calculation and evaluation of the most optimal moves. After initialization, the get_q_value simply returns the value of taking an action in a particular state. Def choose_action allows selection of a move based on the current policy, whether PRANDOM, PGREEDY, or PEXPLOIT. Def update_q_value updates the q value based on the selected algorithm.

We define the list of actions as North/South/East/West/Pickup/Dropoff. Def qtable_to_matrix creates the Q-table in a matrix format for output. Def snapshot_qtables provides a snapshot where if save_prefix is set to true, then the Agent female and male will be saved to two separate .csv files. Def any_dropoff_reached_5 will simply return if any dropoff stations are filled at 5 blocks. Finally, def run_experiment is the main logic for executing the experimentation, defining and initializing many variables to keep track of the current state in the scope of the overall process, and while the current_step is less than the total steps defined in the call to the function, it executes the logic of employing a particular policy in the PDWorld, seeking to have the agent move blocks from the pickup locations to the dropoff tiles until the dropoff is full with 5.

We experimented with different approaches to the PDWorld problem, with our initial approach to the PD World problem of exploration and exploitation consisting primarily of 4 main classes: class PDWorldEnv for establishing the world and the possible things in the world, class TabularQ for determining the desired/optimal action using the best_action function as a way of evaluating the best choice using Q table, class AgentConfig for establishing the parameters for the agent's exploration of the world, and class IndependentLearner for defining the set of 4 different experiments of different approaches to be able to evaluate the differences in performance based on different setups such as using PRANDOM, RGREEDY, PExploit and different warmups. In reinforcement learning, there is typically an inherent tradeoff between exploration of the world and exploitation of the world, wherein exploration means trying new things/exploring spaces and making discoveries, whereas exploitation means choosing the optimal choice that is known to be desirable.

The class PDWorldEnv sets up the rules of the world the agents are to explore, with 10 main functions defined. Def reset allows the world to be reset to its initial state. Def is_terminal checks if the world is in its finished state. Def total_remaining shows the total amount of objectives that are still in their initial bucket. Def bucket remaining shows the number of objectives that are still on a particular tile in their initial bucket. Def get state shows where an agent is and the condition that the agent is in, such as if it's carrying, turned, etc. def other_agent does something. Def _move is the function that calls the movement of the agent, if not North then South, if not South then West, and if not West then East. def step is what controls the actions of the agent, first selecting the direction from the previous move function, then choosing PICKUP or DROPOFF if appropriate, and also assigning an appropriate reward if applicable. Def applicable ops checks the state of the agent to see which operations are even applicable or possible in its current state, finding if moving N/S/E/W is possible, and likewise for the actions of picking up or dropping

off. Def manhattan distance does a simple calculation that is not the euclidian distance but instead rather than calculating the hypotenuse of the distance between two points, it calculates the x distance and the y distance and simply adds (the absolute value of) them together. The class TabularQ is for the Q table, which, after initialization, has def best_action, which has the logic for how the desired action is determined through its calculation of maximizing the Q function. Def update q updates the q table for the action based on the q of the action, the alpha, and the target.

Analysis:

All three setups for experiment 1, the setup for experiment 2, both setups for experiment 3, and the setup for experiment 4 all had a similar characteristic of the cumulative reward plot, looking highly similar in shape, with the cumulative reward decreasing in a mostly linear fashion across the entirety of the steps taken by the agent. Given this was true across all setups, it is highly reflective not of the viability of any one approach to exploring and exploiting the space, but is clearly reflective of the setup of the task itself; the constant -1 pain that the agent feels every time it takes a move is such that regardless of the approach employed across these 4 experiments, the reward of +13 for a successful pickup/dropoff is insufficient to offset the cost of exploring the space with the -1 for every single North/South/East/West move taken. The agent cannot be reasonably expected to perform pickups/dropoffs efficiently enough for a positive cumulative reward, because the cost of exploration is too high relative to the value of successfully performing a dropoff.

That being said, however, experiment 1c vastly outperformed 1a and 1b in the degree of negativity for its cumulative reward, with a negative value that was less than half of the previous 2 setups. This result tracks tightly with what we would expect conceptually from the approach of 1c: employ PRandom until 500 steps, then PGreedy after the 500th step. The reason for this is an alignment of what these different approaches to policies actually mean: PRandom will randomly choose the movement for the next action, PGreedy will always choose the action that has the highest q value at that moment, and PExploit will choose the highest q value (PGreedy) with an 80% chance, and will randomly move in the space (PRandom) the other 20% of the time. With PRandom, exploration of the space is the emphasis, whereas with PGreedy, simply selecting the calculated best local option for that particular move is the choice, and critically PExploit refuses to be exclusively one strategy or the other, instead employing an emphasis on choosing the best move at that step, but also leaving a 20% chance of simply moving in the space. This means that PRandom will focus more on exploring the area using North/South/East/West operators, whereas PGreedy will focus more not on discovering new things/areas in the space but instead capitalize on what it already knows will add to the cumulative reward, namely the picking up and dropping off of objective blocks, with PExploit somewhat in between the two strategies by employing each of them with an 20% or 80% likelihood respectively. As such, it is aligned with our conceptual understanding of these strategies that a pure PRandom exploration of the space for the entirety of the 8000 steps would be an ineffective strategy and result in a poor cumulative reward, by not ever being guided by a policy to maximize the q value of the reward at that step. Interestingly, however, is that the cumulative reward is not markedly better for the strategy that

uses PRandom for the initial 500 steps and then relies on PGreedy for all steps after 500. Surprisingly, there is no benefit to choosing the optimal q-value move, even after using the first 500 steps to explore the space. The optimal strategy for our overall setup across experiment 1 is to use PRandom to explore the space for the initial 500 steps, then use the hybrid approach present in PExploit to have a higher weighted likelihood of employing PGreedy 80% of the time, but not rigidly restricting to that strategy, allowing space for the use of PRandom 20% of the time. As for the quality of coordination between the agents for exp 1b and 1c, we see that the average Manhattan distance between the two agents was 3.22, for both experiments. This shows that both did relatively well in keeping to themselves, considering the grid size is 5. This type of coordination between agents is similar to all our other experiments, even with different policies, it seems that the agents learn to work and synchronize to each other's work pretty well over time.

For experiment 2, Q-learning outperforms SARSA broadly, and that becomes clear from episode 8 on, with total rewards for Q learning exceeding that of SARSA, the former of which gets decidedly positive by the final episodes, whereas SARSA barely squeaks out a single positive in total rewards, except in the very last episode quite narrowly. As for agent coordination, we can see that the average distance between our agents throughout our 8000 steps is about 3.23, which we would say is quite good, considering we have a grid size of 5. This doesn't mean that our agents didn't run into each other at times, or get quite close to each other, which they probably did, but on average, our agents learned to avoid each other and work towards getting to the terminal state.

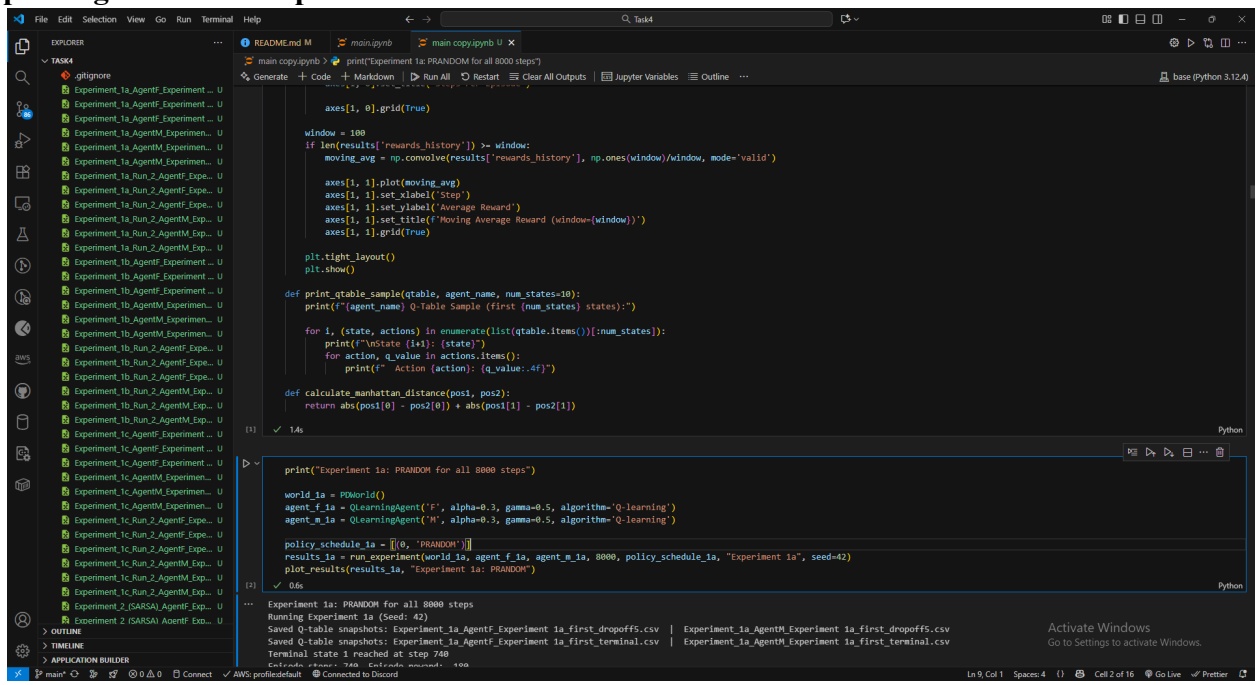
For experiment 3, Q-learning with an alpha of 0.45 outperformed Q-learning with an alpha of 0.15 with respect to the measurement of the cumulative reward. The reasoning for this is likely that an alpha of 0.15 was too low a learning rate for the agent to successfully adjust its approach to account for new information to be highly effective. This is reflected prominently in two of the additional charts outside of just the pure cumulative reward. The steps per episode plots for the two setups reflect the fact that the lower learning rate with an alpha of 0.15 had 3 episodes with the number of steps approaching 1000; alternatively, the alpha of 0.45 had a single one. Additionally, the entire distribution and scale of the total rewards per episode plot is highly reflective of the reality that 0.15 was likely too low of a learning rate to be optimal in maximizing reward in the PDWorld; the alpha of 0.45 has a scale that shows the values at best approaching a flat 0, which is desirable for the total rewards given that all values for both plots were negative (which relates to the reality that the structure of this problem makes a net positive cumulative reward not feasible for the agents given the cost of moving exceeds the broad value of +13 to accomplish the pickup/dropoff). Alternatively, the total rewards per episode plot for the alpha of 0.15 is worse for both the peak and trough for any single episode, with the total rewards never approaching a flat 0 enough for the scaling to even show a 0. Taking a step back and evaluating this broadly, it is likely the case that a higher learning rate than 0.15 was preferable because adjusting to the context of the PDWorld requires a more rapid response to the experience and information than 0.15 was resulting in. For context, the best performing was actually the alpha of 0.30 baseline, with not only the highest cumulative reward but also having the only cases of a positive cumulative reward across all 3 alpha values. Q learning is off-policy, and the next action being employed here is decided based on the best action in a set of actions, whereas SARSA determines the action strictly by the policy.

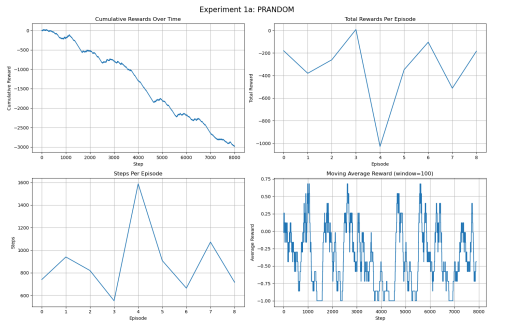
Experiment 4's results are reflective of the fact that the agent was positively responsive to the pickup location changing after 3 episodes. This conceptually aligns with the modern machine learning concept of overfitting: when a model gets really strong at one particular situation/environment/setup but then fails to extrapolate what it has learned in different contexts. The plot for our experiment 4 illustrates that the agent and approach are somewhat adaptive, and in that respect is not overfit to a particular situation (it may not be a 1:1 conceptual translation, but the general idea of rigidly performing well with only a particular setup is a concept that is relevant to both experiment 4's reinforcement learning and modern neural networks overfitting to a particular training set but failing to extrapolate broader lessons), because once the pickup locations changed and the agent could no longer find the same rewards in the same locations, it was able to adapt, reflected clearly in the high number of total rewards show in the plot following the movement of the pickup locations, even getting a positive total rewards on the 9th episode. Note, we found out later that we are only supposed to run until the 6th terminal state. Keeping this in mind, if we look at our graphs and stop at 6, we see that our agents still perform pretty well after the change, as our "peaks and valleys" seem to be higher in reward. This is in reference to the "Experiment 4: Adaptation to Environmental Changes". You can get a closer look at our code file. Also, our average Manhattan distance between our agents is 3.39, which is quite good and also better than our other experiments. Keep in mind our board size is 5, so this is a decent average gap between the 2 agents. This supports the fact that they were coordinating quite well.

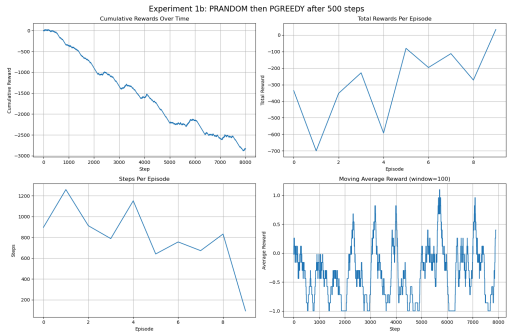
Overall, we think all of the experiments did pretty well, with some slight differences between them. 1c seems to have done the best, as its net reward is the most "positive" out of all of our experiments, then we have a close second in Experiment 3 with an alpha of 0.30. As for our Q-tables, any path that has a reward of zero is profitable for our agent. This is because just moving the agent means -1 reward, so if we can even it out, then that's at least somewhat good! This is true for all of the Q-tables we report. Also, in general, all of our experiments agree on the fact that cumulative rewards do decrease over time, and that the net reward in an episode across all experiments seems to be negative. Where they do disagree is how "negative" the rewards remain, so some of the experiments you will see that the net reward in a particular episode goes above zero, which is good, but then some experiments remain having a net negative reward throughout, though some are worse than others.

Note: Evidence of our code running is found in the ipynb file itself; the screenshots here are the proof. If you want to verify, you can do so by running it again. Here is a screenshot

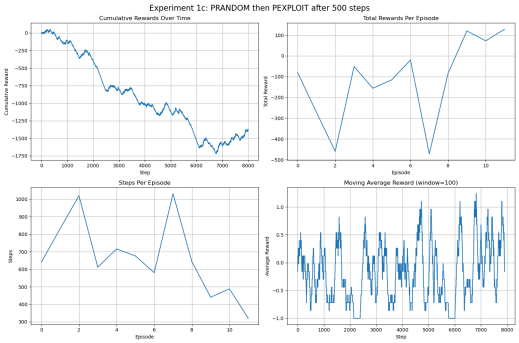
proving the code compiles:



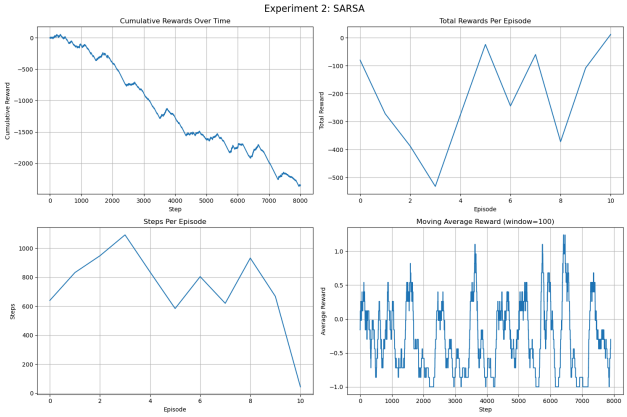




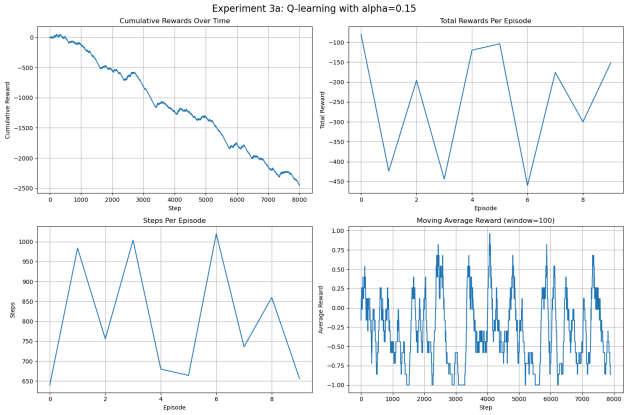
Experiment 1c: PRANDOM then PEXPLOIT after 500 steps



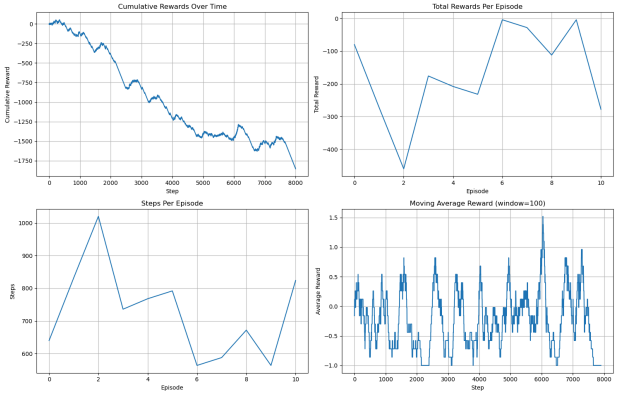
Experiment 2: SARSA

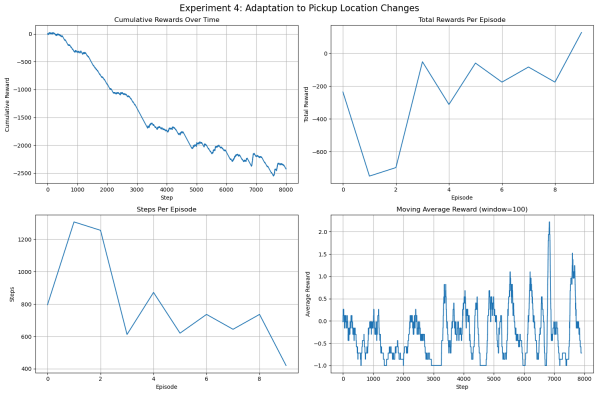


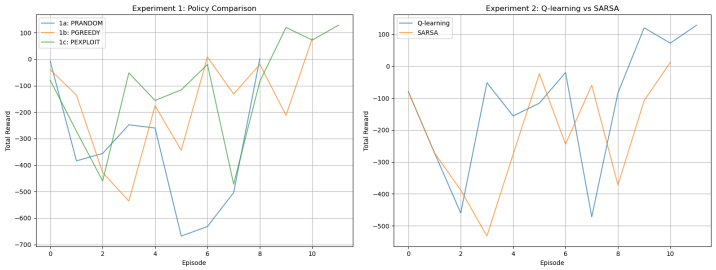
Experiment 3a: Q-learning with alpha=0.15

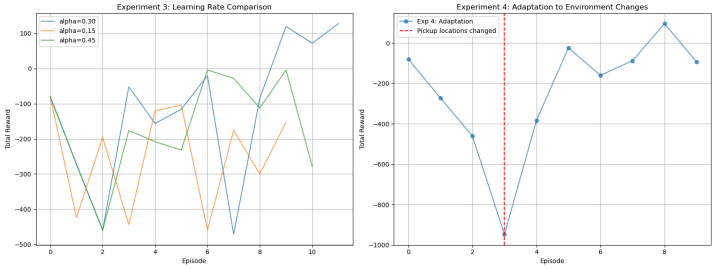


Experiment 3b: Q-learning with alpha=0.45




Experiment 4: Adaptation to Pickup Location Changes
Cumulative Rewards Over Time
Cumulative Reward
Step
Total Rewards Per Episode
Total Reward
Episode
Steps Per Episode
Steps
Episode
Moving Average Reward (window=100)
Average Reward
Step


Experiment 1: Policy Comparison
1a: PRANDOM
1b: PGREEDY
1c: PEXPLOIT
Total Reward
Episode
Experiment 2: Q-learning vs SARSA
Q-learning
SARSA
Total Reward
Episode


Experiment 3: Learning Rate Comparison
alpha=0.30
alpha=0.15
alpha=0.45
Total Reward
Episode
Experiment 4: Adaptation to Environment Changes
Exp 4: Adaptation
Pickup locations changed
Total Reward
Episode