DA6400 Programming Assignment 3

Team:

Rupankar Podder *CS24S008*
Manoj Kumar CM *DA24S018*

Github Link: https://github.com/2coolcoder/SMDP_Taxi_Environment

SMDP Q learning:

SMDP QLearning Update:

$$Q(S, O) = Q(S, O) + \boldsymbol{\alpha}[r + \boldsymbol{\gamma}^{\boldsymbol{\tau}} max_{O'} Q(S', O') - Q(S, O)]$$

In the proposed SMDP Q Learning approach for the Taxi-v3 problem, 4 options are defined:

1. **Goto R** : Move to **Red** position, and pick/drop the passenger depending on whether the passenger is in the car.
2. **Goto G** : Move to **Green** position, and pick/drop the passenger depending on whether the passenger is in the car.
3. **Goto Y:** Move to **Yellow** position, and pick/drop the passenger depending on whether the passenger is in the car.
4. **Goto B** : Move to **Blue** position, and pick/drop the passenger depending on whether the passenger is in the car.

Each of the options follows its own policy to reach its desired destination.

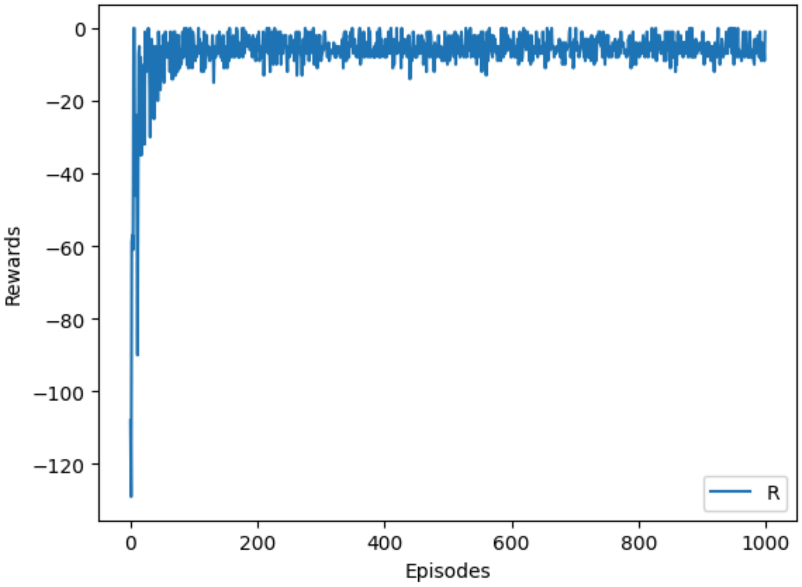
Each of the options learn a policy based on a simple Q learning approach instead of this policy being deterministically defined such that the agent learns to navigate the taxi to the desired location/ destination i.e., (Red/ Blue/ Green/ yellow). Within the policy the agent takes on the primitive action (North/ South/ East/ West).

State Space: taxi_row x taxi_column x passenger_location (whether in the taxi)
Action Space: All 4 primitive actions
Q-Table Dimension: 5 x 5 x 4 (taxi_row x taxi_col x 4 navigation actions)

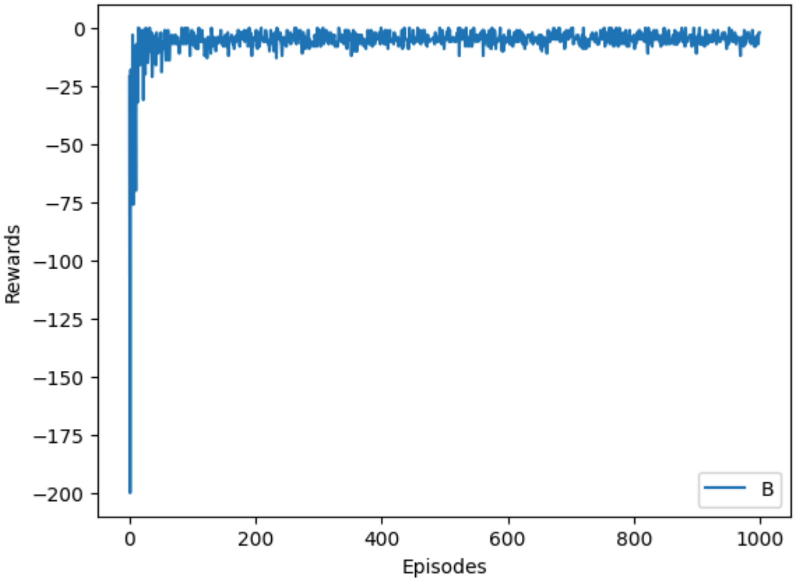
We are separately handling the picking/ dropping based on passenger location. That is why we have not included them in the Q-table. This allows faster learning due to a smaller Q-table.
We are learning the options policy using **Q-Learning**.
While learning the policy, the options get a **reward of -1 for every step**, until reaching the goal (reaching the start/ destination and picking/ dropping the passenger).

Reward plots for each of the Options who policy is learnt via Q learning:

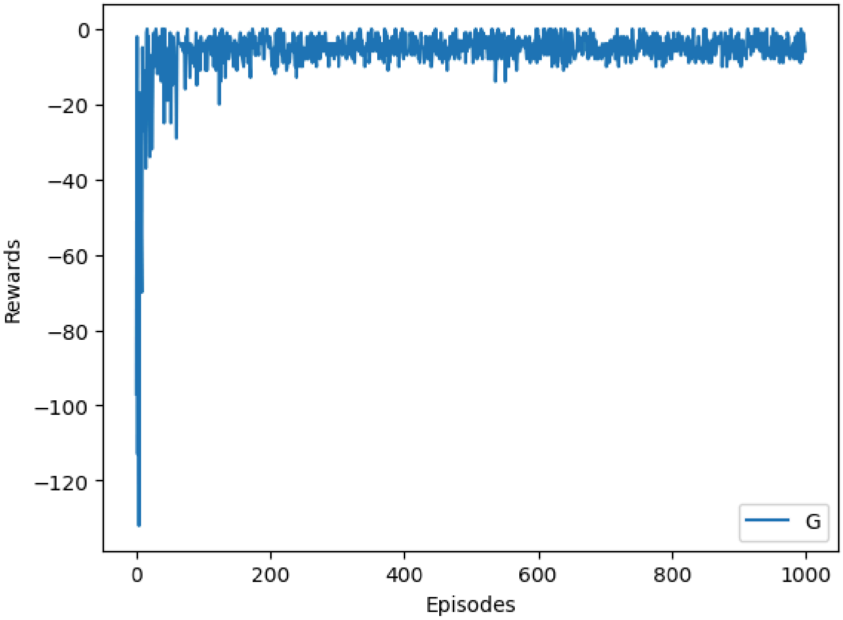
Goto R



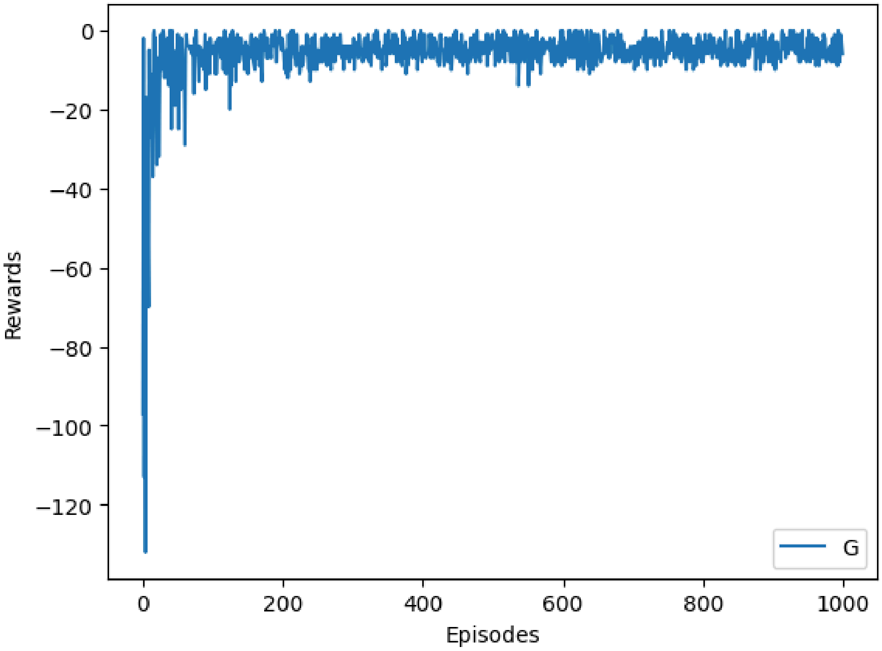
Goto B



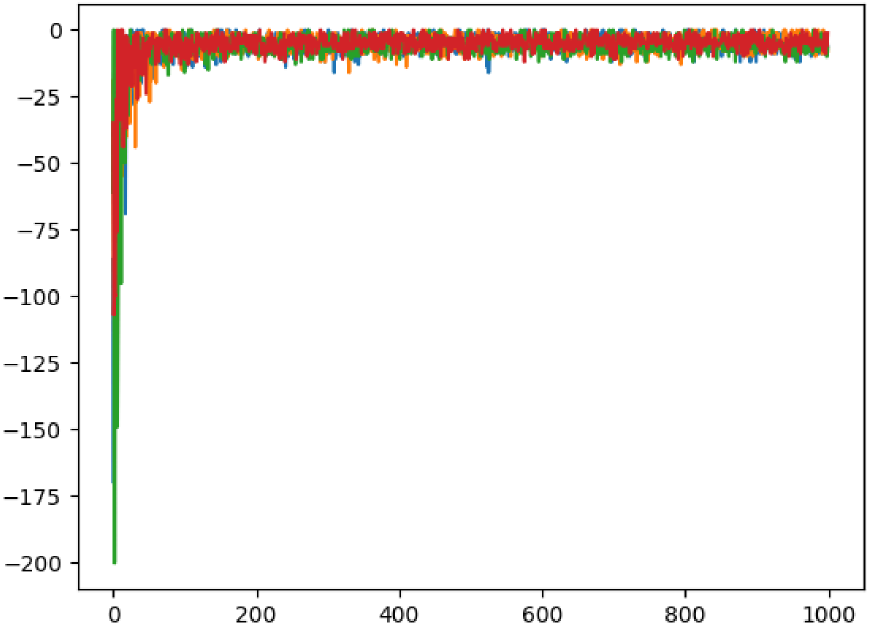
Goto G


0
−20
−40
−60
−80
−100
−120
Rewards
0
200
400
600
800
1000
Episodes
G

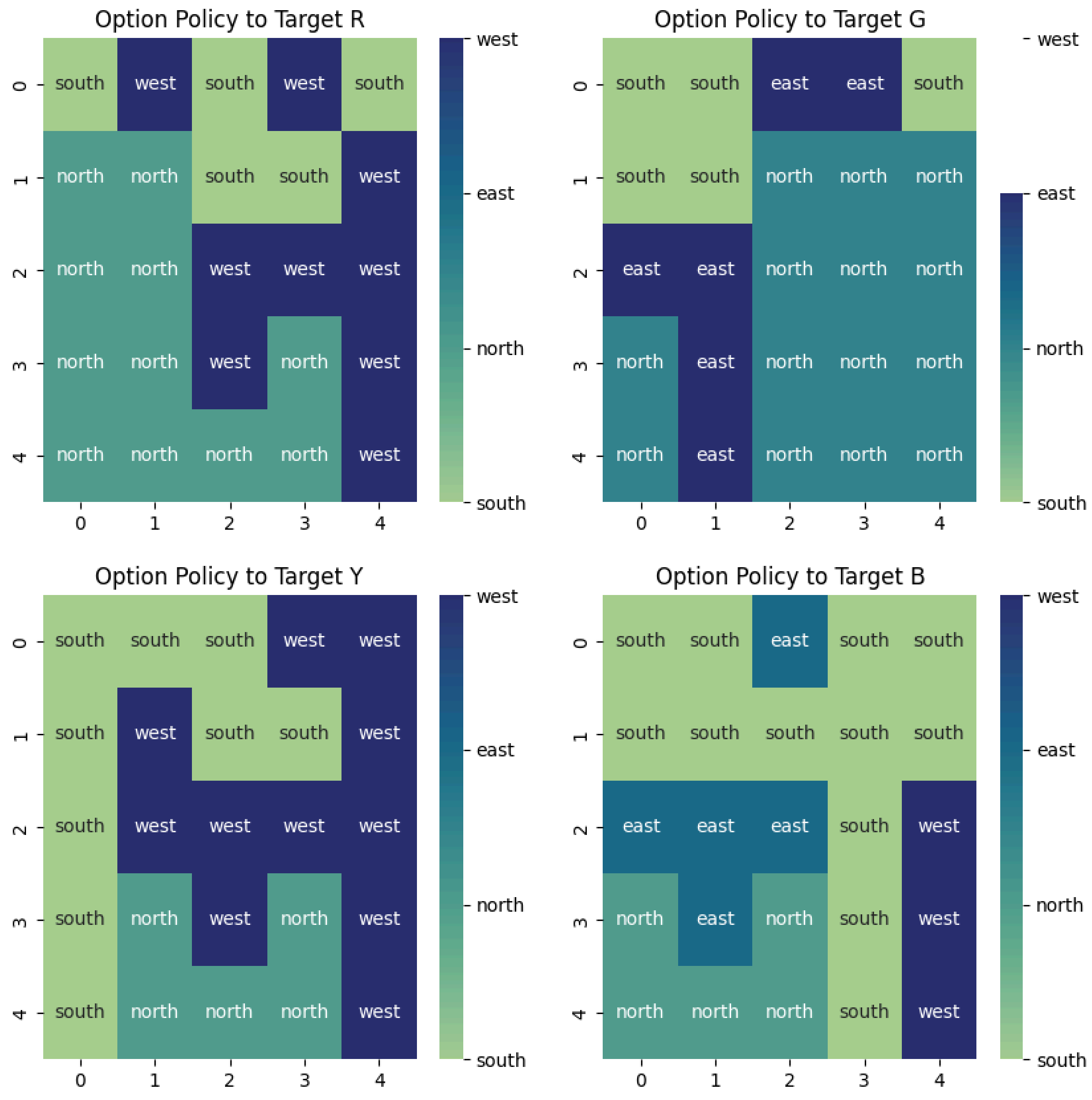
Goto Y


0
−20
−40
−60
−80
−100
−120
Rewards
0
200
400
600
800
1000
Episodes
G

The below graph shows the **episodes vs return graph of all the 4 options**.



Learned Options:



Option Policy to Target R

	0	1	2	3	4
0	south	west	south	west	south
1	north	north	south	south	west
2	north	north	west	west	west
3	north	north	west	north	west
4	north	north	north	north	west

Option Policy to Target G

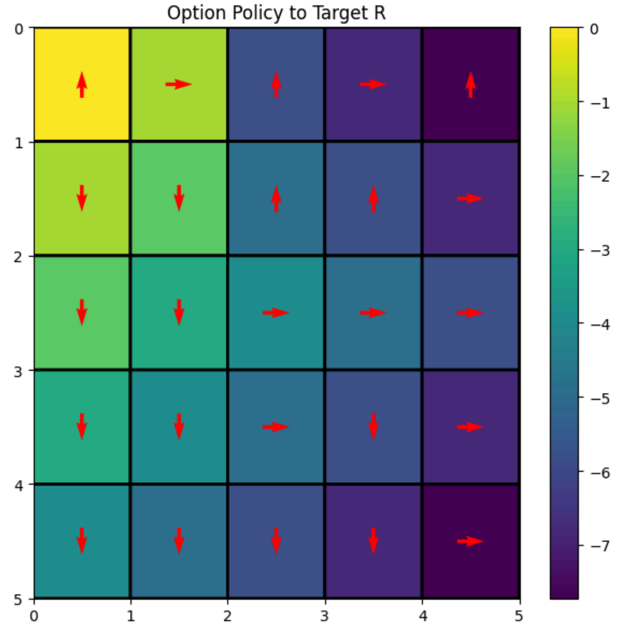
	0	1	2	3	4
0	south	south	east	east	south
1	south	south	north	north	north
2	east	east	north	north	north
3	north	east	north	north	north
4	north	east	north	north	north

Option Policy to Target Y

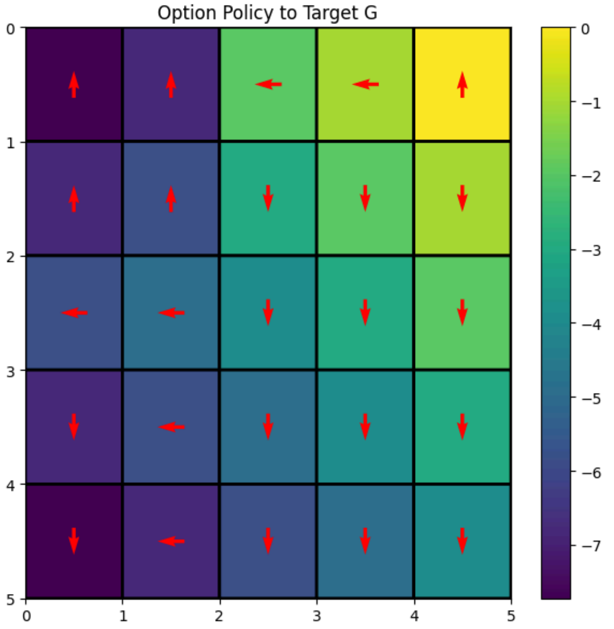
	0	1	2	3	4
0	south	south	south	west	west
1	south	west	south	south	west
2	south	west	west	west	west
3	south	north	west	north	west
4	south	north	north	north	west

Option Policy to Target B

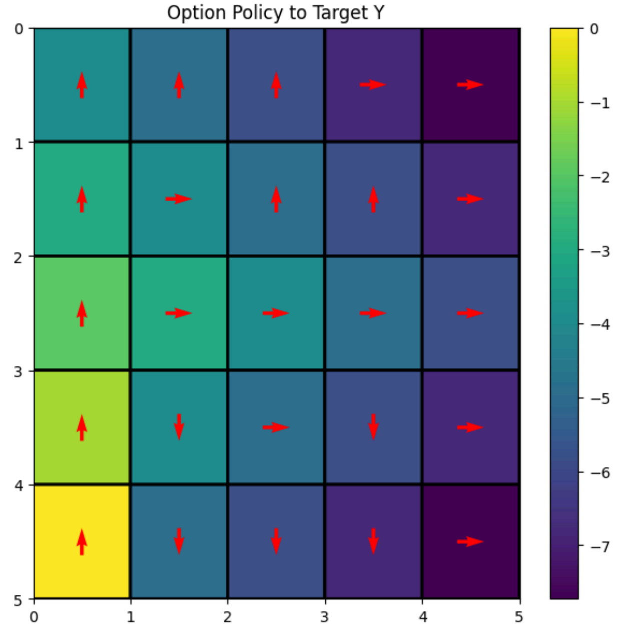
	0	1	2	3	4
0	south	south	east	south	south
1	south	south	south	south	south
2	east	east	east	south	west
3	north	east	north	south	west
4	north	north	north	south	west



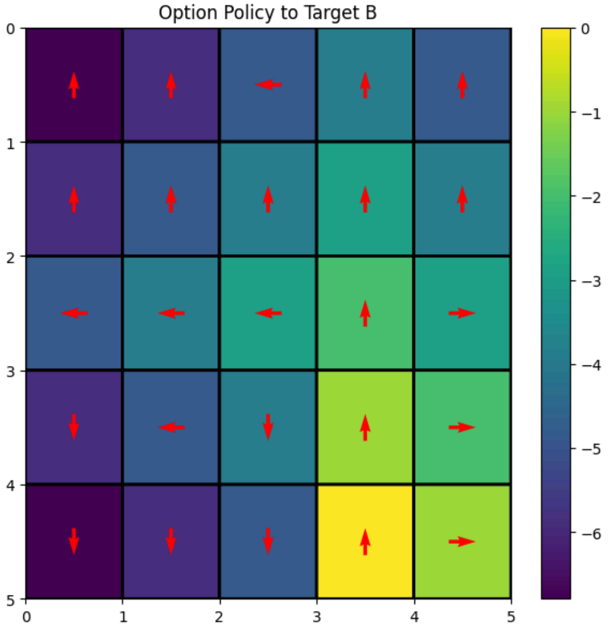
Option Policy to Target R



Option Policy to Target G



Option Policy to Target Y



Option Policy to Target B

Note:

↑ corresponds to south
↓ corresponds to North
-> corresponds to West
<- corresponds to EAST

Policy Explanation:

Each of these learnt policies follows the **shortest path** to the corresponding locations (0: red, 1: green, 2: yellow, 3: blue) from any other location in the grid.

Upon reaching the target location,

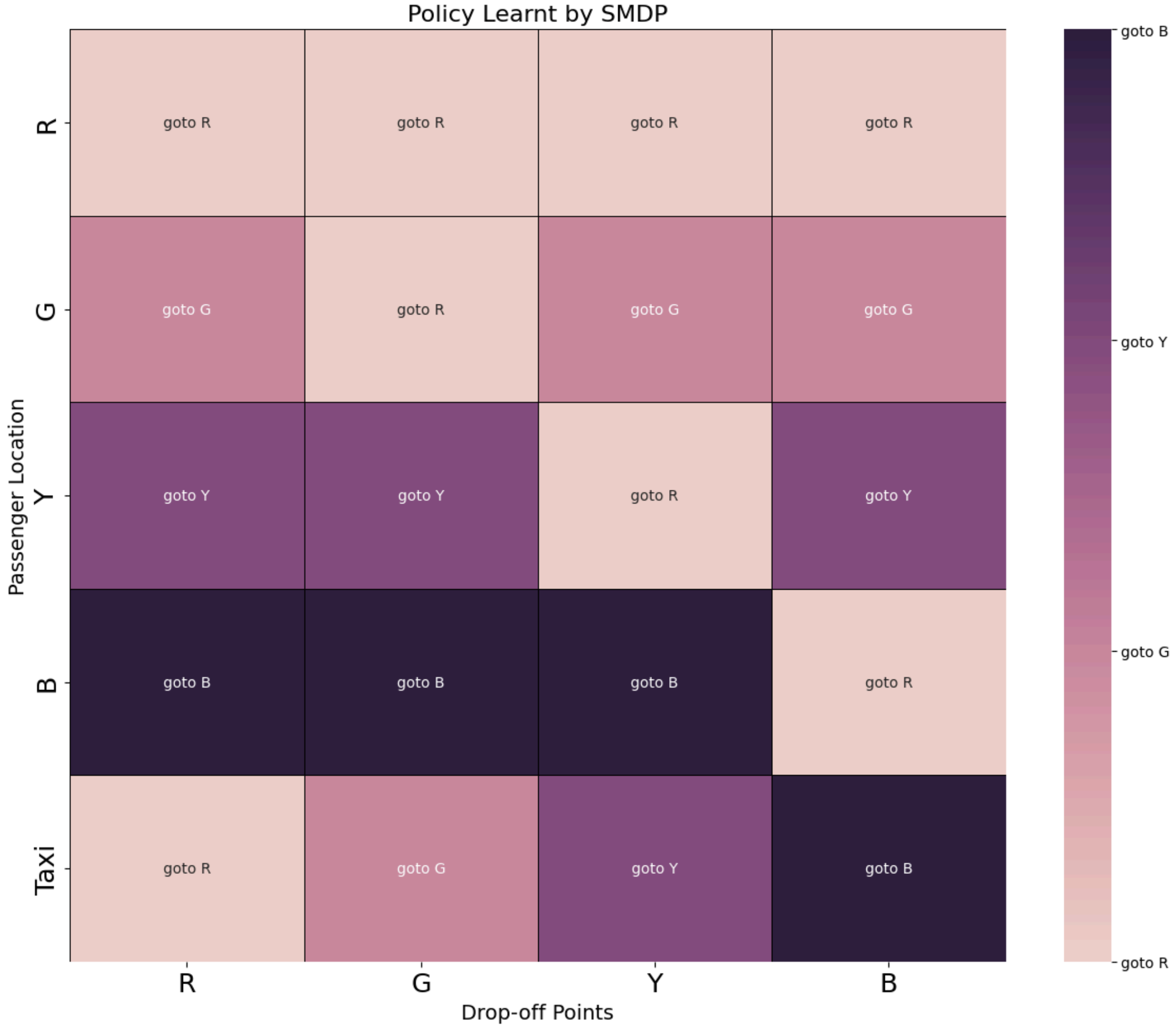
- If passenger_location < 4 : Pick up the passenger
- If passenger_location == 4 : Drop the passenger

SMDP

State space: (passenger location, passenger destination)
Action Space: 4 options
Q-Table Dimension: 5 x 4 x 4 (passenger_location x destination x options)

When the SMDP performs an option, we calculate the number of steps and discounted return that is used to update the SMDP Q-value.

Learned Policy:



The learned policy can be summarised as:

- If the passenger is not in the taxi, run the corresponding action to go to the passenger location and pick up the passenger. The option will know to pick up the passenger (not drop) from the passenger location (not in taxi i.e. <4)

- If the passenger is in the taxi, run the corresponding action to go to the destination. The option will drop the passenger (not pick) because the location of the passenger is in the car (=4)

This is the expected policy, that runs the correct options to pick up the passenger and drop the passenger.

2. Different set of Options:

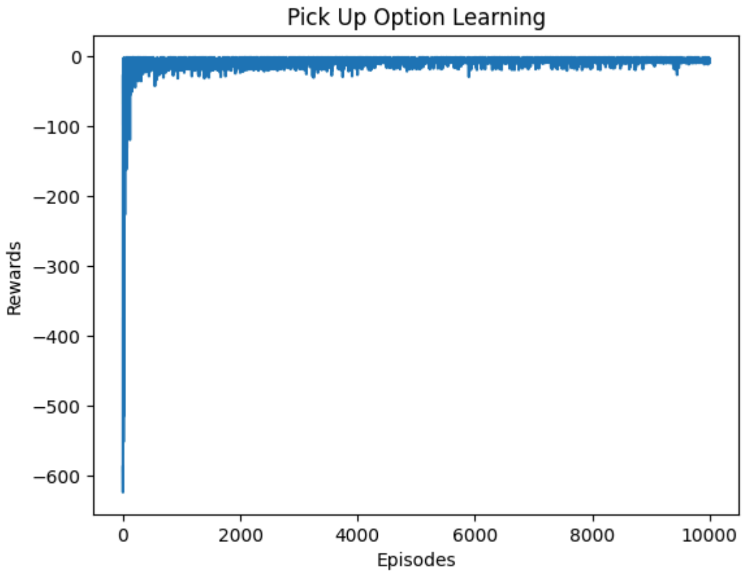
Two Options which is mutually exclusive is defined:

1. Pick_option
    a. Navigates to the passenger and picks up the passenger
    b. Can be called from any state
    c. Terminates when the passenger is in the taxi
    d. Does nothing if the passenger is already in the taxi
2. Drop_option
    a. Navigates to the destination and picks up the passenger
    b. Can be called from any state
    c. Terminates when the passenger is dropped off in the correct location
    d. This option fails if the passenger is not in the taxi at the start.

Pick Option:

State Space: passenger_location x taxi_row x taxi_column
Action Space: All 6 actions
Q-Table Dimension: 4 x 5 x 5 x 6 (psg_location x taxi_row x taxi_col x 6 actions)

The option policy is learnt based on the Q Learning approach. Rewards are unchanged, and the option terminates when the objective is achieved (passenger is in the taxi).
The Agent learns to reach the passenger in a small number of steps and picks the passenger and terminates the option; The learning of the agent is given below:

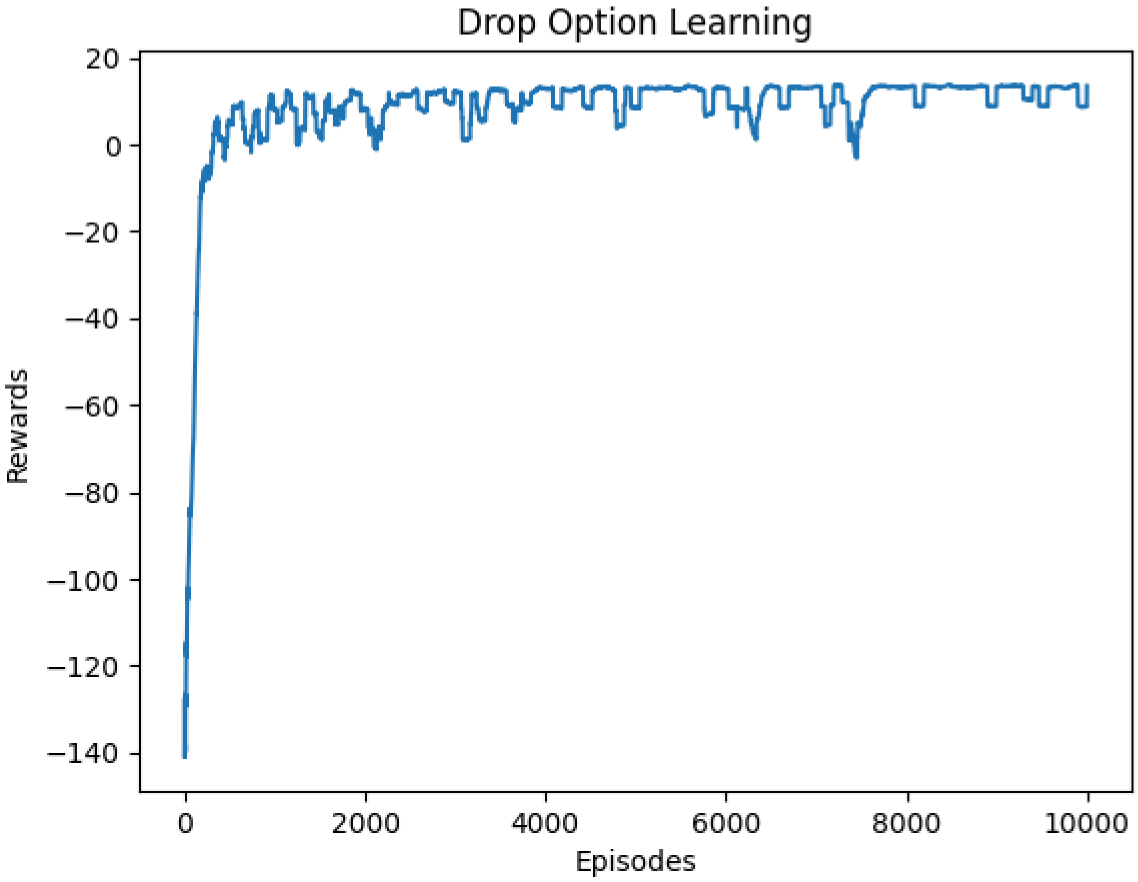


Drop Option:

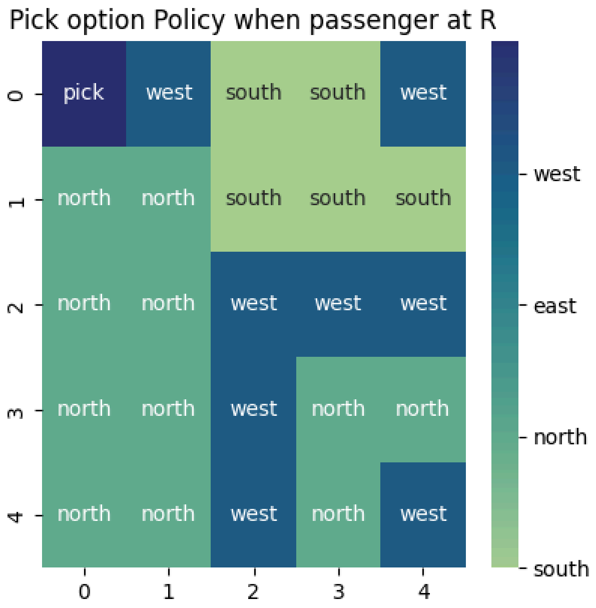
State Space: destination x taxi_row x taxi_column
Action Space: All 6 actions
Q-Table Dimension: 4 x 5 x 5 x 6 (destination x taxi_row x taxi_col x actions (navigation + interaction).

The option is learnt based on the Q Learning approach. The option terminates when the passenger is dropped off at the destination (same as when the episode also terminates). Rewards are unchanged.
While training, the environment is reset and the pick option is used to pick the passenger. Then the Drop option policy is followed and learned.

The Agent learns to reach the destination in small number of steps and drops off the passenger and terminates the option; The learning of the agent is given below:

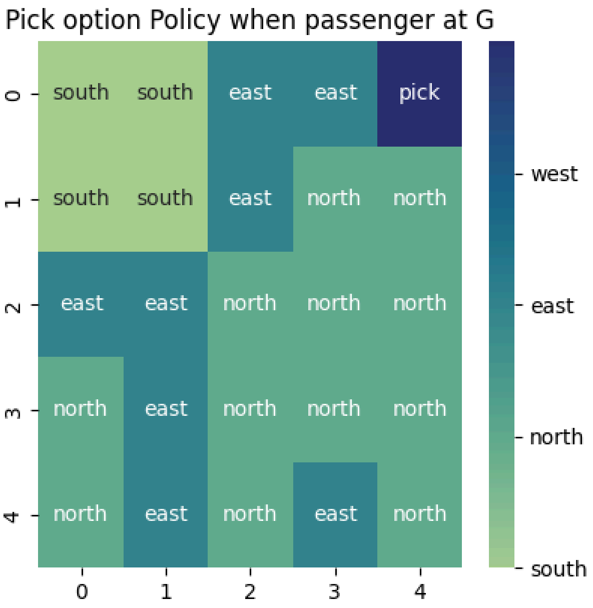


The **Pick Up Option policy** defines the action over all the passenger location, and taxi location. We can visualise the policy for each passenger locations:



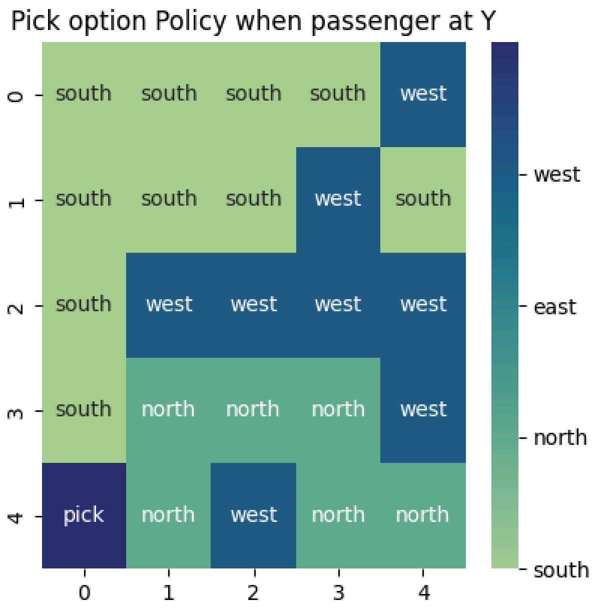
Pick option Policy when passenger at R

	0	1	2	3	4
0	pick	west	south	south	west
1	north	north	south	south	south
2	north	north	west	west	west
3	north	north	west	north	north
4	north	north	west	north	west



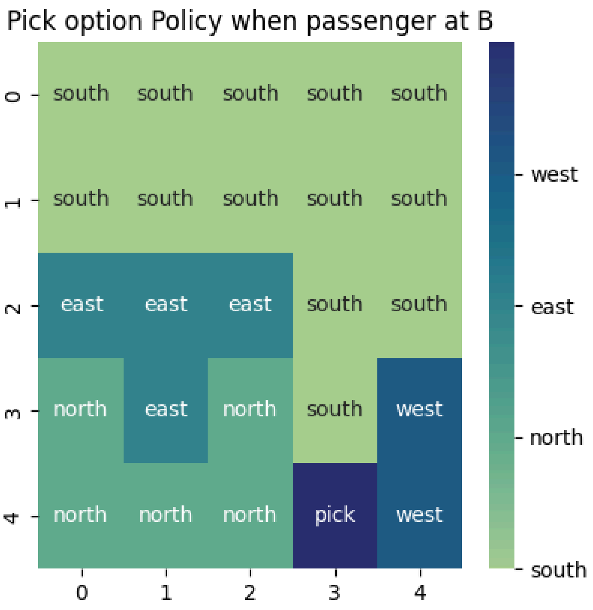
Pick option Policy when passenger at G

	0	1	2	3	4
0	south	south	east	east	pick
1	south	south	east	north	north
2	east	east	north	north	north
3	north	east	north	north	north
4	north	east	north	east	north



Pick option Policy when passenger at Y

	0	1	2	3	4
0	south	south	south	south	west
1	south	south	south	west	south
2	south	west	west	west	west
3	south	north	north	north	west
4	pick	north	west	north	north

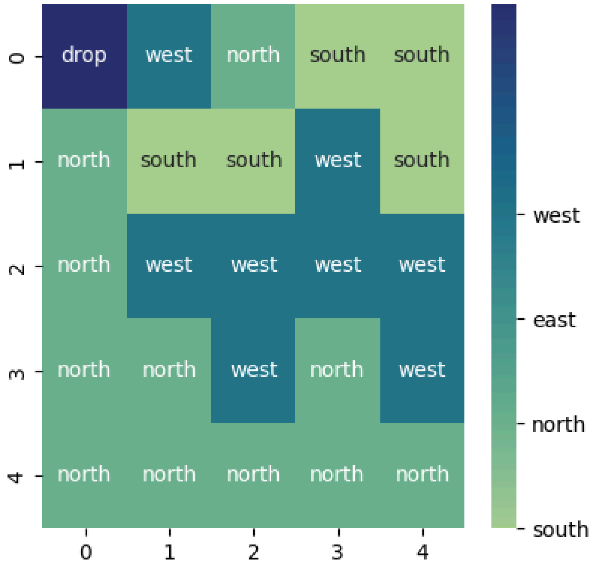


Pick option Policy when passenger at B

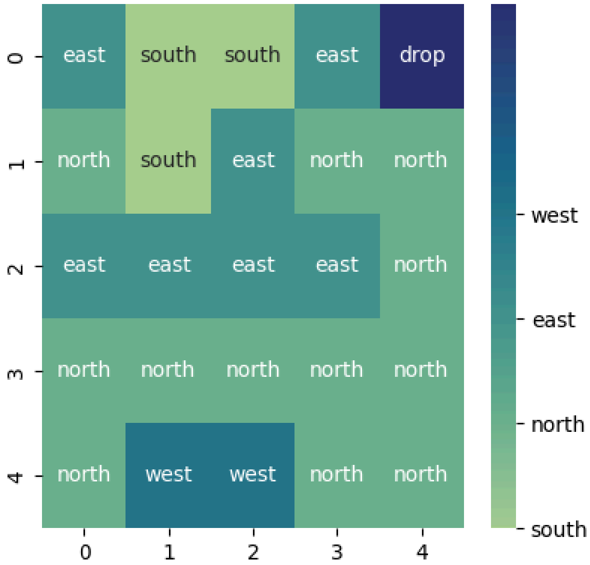
	0	1	2	3	4
0	south	south	south	south	south
1	south	south	south	south	south
2	east	east	east	south	south
3	north	east	north	south	west
4	north	north	north	pick	west

Drop-off Policy:

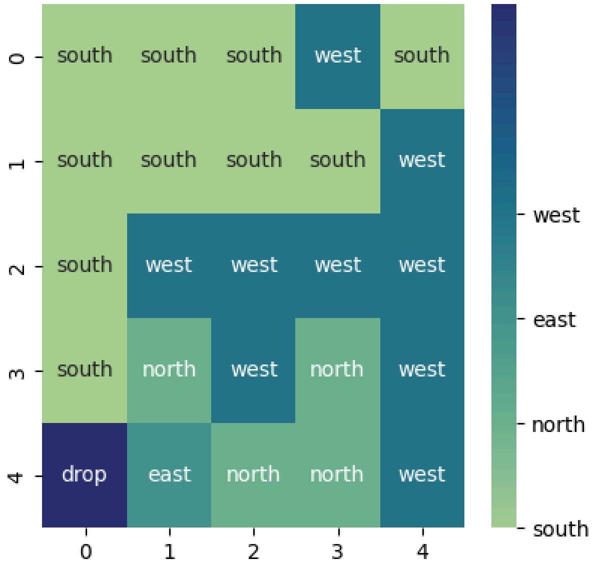
Drop option Policy when destination is R



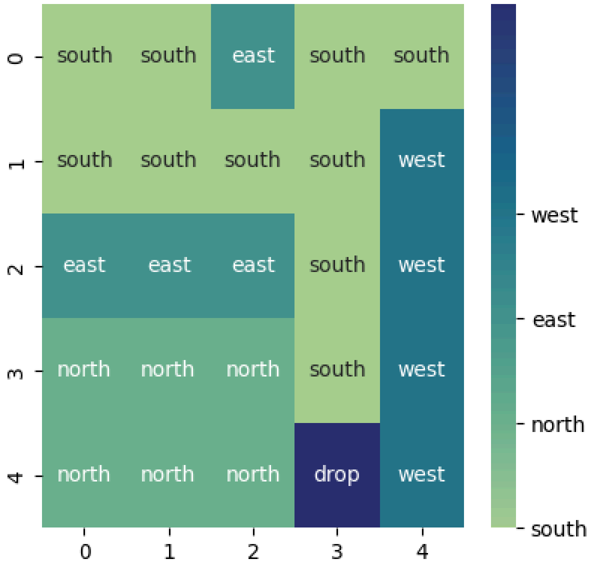
Drop option Policy when destination is G



Drop option Policy when destination is Y



Drop option Policy when destination is B

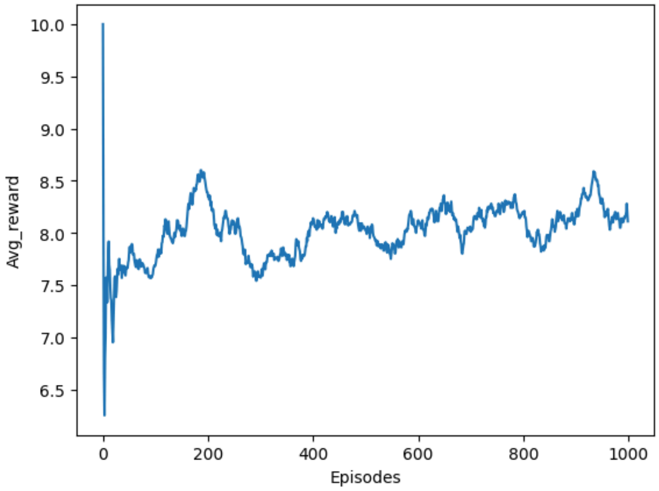


SMDP

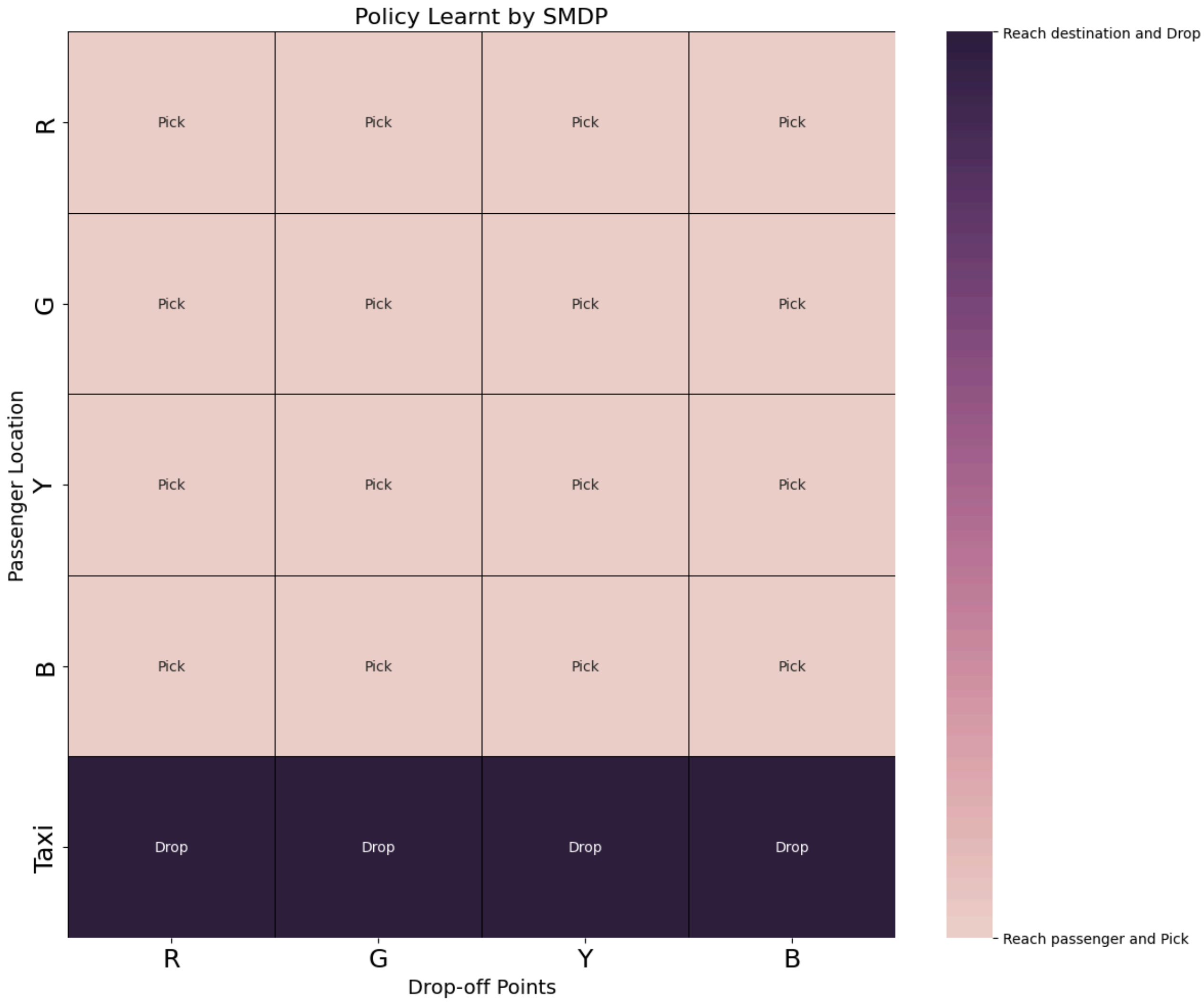
State space: (*passenger_location, destination*)
Action Space: 2 options
Q-Table Dimension: 5 x 4 x 2 (passenger_location x destination x options)

When the SMDP performs an option, we calculate the number of steps and discounted return that is used to update the SMDP Q-value.

Rewards plots for SMDP learning over options:



Policy Learnt by SMDP QLearning:



	R	G	Y	B
R	Pick	Pick	Pick	Pick
G	Pick	Pick	Pick	Pick
Y	Pick	Pick	Pick	Pick
B	Pick	Pick	Pick	Pick
Taxi	Drop	Drop	Drop	Drop

The policy can be summarized as follows:
When the passenger is not in the taxi, the SMDP learns to use the Pick-option and when the passenger is in the taxi, the SMDP learns to use the Drop-option. This is the expected policy, which picks up the passenger and drops them off at the destination.