Description of all Environments

1 LunarLander

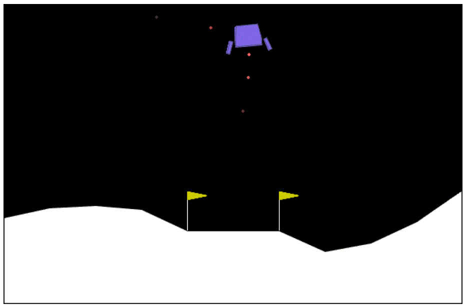

Figure 1: LunarLander

State Space : The state space is a 8 dimensional vector consisting the coordinates of the lander in x and y, its linear velocities in x and y, its angle, its angular velocity, and two booleans that represent whether each leg is in contact with the ground or not.

Action Space : There are four discrete actions available: do nothing, fire left orientation engine, fire main engine, fire right orientation engine.

Reward : Reward for moving from the top of the screen to the landing pad and coming to rest is about 100-140 points. If the lander moves away from the landing pad, it loses reward. If the lander crashes, it receives an additional -100 points. If it comes to rest, it receives an additional +100 points. Each leg with ground contact is +10 points. Firing the main engine is -0.3 points each frame. Firing the side engine is -0.03 points each frame. Solved is 200 points.

Goal : The goal of the Agent is to strategically accelerate the lander to land smoothly on the landing pad without crashing.

Seeds	**RYB-DQN-Exp**	**DQN**
219	299	500
4065	202	334
987	500	176
434	137	260
4218	326	500
846	139	201
2647	349	300
4283	180	341
1190	214	189
4372	189	399
Average	**253.5**	**320.0**

Table 1: Convergence Results

Good episode : We initialize a variable called ”neural_epi” to zero. Whenever the total return of an episode equals or exceeds the value of ”neural_epi,” we add the entire episode to the ”good_episodes” buffer. Additionally, we train the neural network using both the ”good_episodes” buffer and the current episode. Subsequently, we update the value of ”neural_epi” to the total return of the current episode.

2 MountainCar

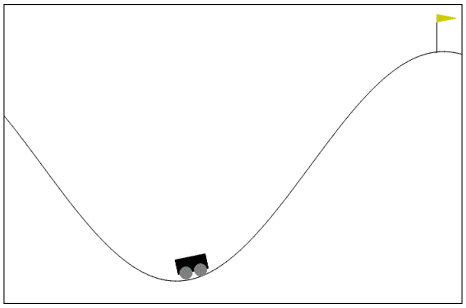

Figure 2: MountainCar

State Space : The state space is a ndarray with shape $(2,)$ which consists of the car's position and velocity. The position can vary within a certain range.

Seeds	**RYB-DQN-Exp**	**DQN**
219	56	94
4065	81	96
987	76	116
434	111	194
4218	84	117
846	83	300
2647	85	169
4283	89	113
1190	123	145
4372	165	167
Average	**95.3**	**151.1**

Table 2: Convergence Results

Action Space : The action space consists of three discrete deterministic actions: Accelerate to the left, Don't accelerate, and Accelerate to the right.

Reward : The agent receives a reward of -1 for each timestep until the goal position is reached. Upon reaching the goal position, the episode terminates with a reward of 0.

Goal : The goal of the Agent is to strategically accelerate the car to reach the flag placed on top of the right hill as quickly as possible. The episode ends if either the position of the car is greater than or equal to 0.5 (the goal position on top of the right hill) or if the length of the episode reaches 1000 timesteps.

Good episode : We initialize a variable called "neural_epi" to -995. Whenever the total return of an episode equals or exceeds the value of "neural_epi," we add the last 200 steps of the current episode to the "good_episodes" buffer. Additionally, we train the neural network using both the "good_episodes" buffer and the last 200 steps of the current episode. Subsequently, we update the value of "neural_epi" to the total return of the current episode. We select only the last 200 steps of the episode as preceding moves are deemed redundant, aligning with our definition of a "good" episode

3 GridWorld

State Space : Grid-World is a 6*6 grid, Where one goal and six walls are strategically placed to make it difficult for the agent to navigate to the goal

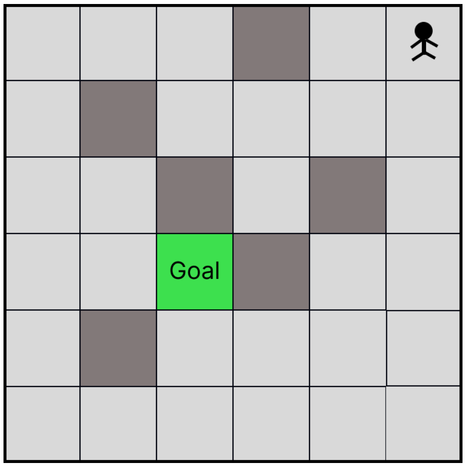

Figure 3: GridWorld

easily, which will be fixed throughout the game.but player's starting position is randomly selected from any empty cell. The state is represented by the coordinates (x, y) of the agent.

Action Space : Player can move to any adjacent cell(if the cell is empty or contains goal) using 4 actions namely LEFT,RIGHT,UP,DOWN.

Reward : The sparse reward setting provides reward of -1 as each intermedeate reward and +100 if player reaches the goal.

Goal : The objective of the game is to navigate the player to the goal state in the least number of steps possible. The episode terminates if the player reaches the goal state or if 50 time steps have elapsed without reaching the goal.

Good episode : Whenever an episode results in reaching a goal state, we refine it by eliminating transitions leading to loops and redundant moves, such as hitting walls and grid boundaries. Subsequently, this refined episode is appended to the "good" replay buffer. The neural network undergoes training using both the current episode and the contents of the "good" buffer. We define an episode as "best" if it doesn't conatain loops and redundant moves, thus justifying the preprocessing step.

Seeds	**RYB-DQN-Exp**	**DQN**
246	166	264
186	94	333
169	158	500
215	241	240
410	268	237
864	295	500
163	317	309
572	280	233
417	315	122
719	143	229
Average	**227.7**	**296.7**

Table 3: Convergence Results