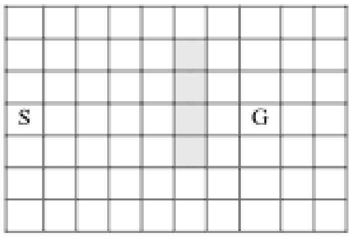
Daniel-Cristian-Marian Țăpuși

MAS Written Assignment

1. Description

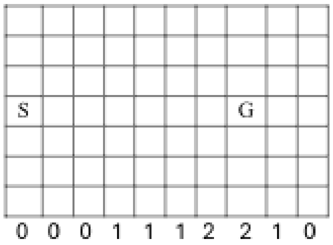
Reinforcement Learning provides a potent methodology allowing agents to determine optimal courses of action by interacting with an environment and utilizing the feedback received. The present document examines the implementation of this learning paradigm for navigation within two dimensional gridworlds, a standard problem for assessing intelligent systems. A key component of this examination involves contrasting three well known algorithms SARSA, Q learning, and Double Q learning regarding their effectiveness in guiding a lone agent between an initial state 'S' and a target state 'G'. The adaptability inherent in these methods was probed using two unique 7x10 grid setups. One setup, designated Gridworld A, contains fixed barriers within the navigational space, whereas the alternative, Gridworld B, introduces environmental complexity through an upward wind current impacting movement across specific columns, forcing the agent to compensate for environmental forces.

Gridworld A



Gridworld B introduces a dynamic element in the form of an upward wind affecting specific columns, forcing the agent to compensate for environmental forces. The agent's learning process involves receiving a negative reward for each step taken and a positive reward upon reaching the goal, aiming to find the most efficient path.

Gridworld B



Understanding how these algorithms adapt performance contingent upon parameter settings formed a core pursuit, involving exploration of different values for the exploration factor epsilon ε and the learning step size alpha α. Key benchmarks were established to measure their success, including the time taken to converge upon a stable operational strategy, the step efficiency characterizing the learned paths determined by averaging step counts, plus the general robustness manifest in consistent performance. The scope of the report also incorporates an examination of a multi agent situation within the Gridworld B context. This involved three separate agents, each guided by Independent Q Learning IQL principles, coordinating efforts toward a simultaneous goal state arrival. A unique reward structure incentivizing synchronized task completion was implemented to underscore the cooperative aspect. Investigating this setup provided grounds to analyze the difficulties related to fostering coordination using standard reinforcement learning approaches when multiple agents interact. The remainder of this document is structured to first elaborate on the methodology employed, then present the comprehensive results from all simulations, and finally discuss the meaning and consequences of these observations.

2. Methodology

The base environment was a 7x10 grid. In both Gridworld A and Gridworld B, the agent started at position (3,1) and aimed to reach the goal at (3,8). Agents could move Up, Down, Left, or Right. Gridworld A contained static obstacles located in column index 5 (the 6th column) at row indices 1 through 4, which the agent could not enter. Gridworld B contained no obstacles but featured an upward wind dynamic. The strength of the wind varied by column, defined by the vector [0, 0, 0, 1, 1, 1, 2, 2, 1, 0]. When an agent took an action, its resulting column position was determined first, then the wind strength corresponding to that column was applied as an upward shift (reducing the row index) before final boundary checks and state transitions were determined.

For the single-agent task (Task 1), three algorithms were implemented:

- **SARSA:** An on-policy algorithm where the Q-value update incorporates the value of the next state and the next action chosen by the policy:
$$Q(s,\ a)\ \Leftarrow\ Q(s,\ a)\ +\ \alpha\ \cdot\ [r\ +\ \gamma\ \cdot\ Q(s',\ a')\ -\ Q(s,\ a)]$$
- **Q-learning:** An off-policy algorithm that updates based on the maximum possible value obtainable from the next state, regardless of the policy followed:
$$Q(s,\ a)\ \Leftarrow\ Q(s,\ a)\ +\ \alpha\ \cdot\ \left[r\ +\ \gamma\ \cdot\ \max_{a'}\quad Q(s',\ a')\ -\ Q(s,\ a)\right]$$
- **Double Q-learning:** Designed to mitigate maximization bias, this algorithm maintains two Q-tables (Q1, Q2). Action selection uses the average value (Q1+Q2)/2. Updates are applied randomly to either Q1 or Q2, using the action selected by one table and the value estimated by the other table:
$$\begin{aligned}Q1(s,\ a)\ \Leftarrow\ & Q1(s,\ a)\ +\ \alpha \\ & \cdot\ \left[r\ +\ \gamma\ \cdot\ Q2\left(s',\ \arg\max_{a'}\quad Q1(s',\ a')\right)\ -\ Q1(s,\ a)\right]\end{aligned}$$

For the multi-agent task (Task 2), Independent Q-Learning (IQL) was implemented using the Q-learning algorithm framework. Three agents operated simultaneously in Gridworld B. Each agent maintained and updated its own independent Q-table using the standard Q-learning update rule, receiving its individual reward and observing its own state transitions. The agents treated each other implicitly as part of the environment's dynamics. The reward structure was specifically designed for cooperation: if all three agents reached the goal state (3,8) in the same time step, each received a reward of +10. If any agent reached the goal but not all did in that time step, each agent received -0.5. If no agent reached the goal, each received -1. An episode concluded only when all three agents were situated at the goal state.

The experiments systematically varied key hyperparameters. The exploration rate, epsilon (ε), was tested at values 0.1, 0.2, and 0.3. The learning rate, alpha (α), was tested at 0.1, 0.3, and 0.5. The discount factor, gamma (γ), was fixed at 1.0 (undiscounted task). Each specific configuration (algorithm, environment, ε, α) was executed for 500 episodes. To ensure reliability and mitigate the effects of randomness, each configuration in Task 1 was run 5 times, and in Task 2 was run 3 times. The results reported (steps per episode, rewards per episode) are the average across these runs. A step limit of 1000 steps per episode was enforced to prevent infinite loops in cases where the agent(s) failed to find the goal.

Performance was evaluated using three primary metrics:

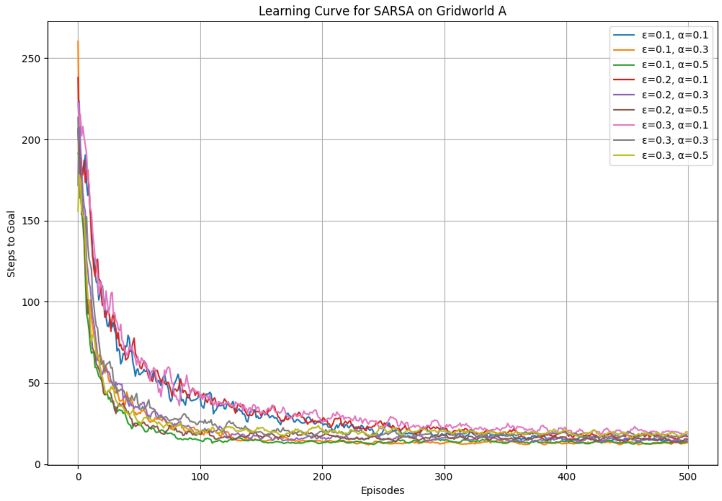
- **Convergence Time:** Estimated as the number of episodes required for the average steps per episode to stabilize, assessed using the detect_convergence function (window=50, threshold=0.05) applied to the averaged learning curves and confirmed by visual inspection of plots.
- **Path Efficiency:** Measured by the average number of steps taken to reach the goal during the later phase of training (calculated over the final 50 episodes of the averaged run). Lower values indicate higher efficiency.
- **Robustness:** Assessed qualitatively based on the stability and low variance of the learning curves in later episodes and the consistency of performance suggested by averaging across multiple runs. The standard deviation of steps in the final 50 episodes was also considered (std_final_steps noted in analysis).

3. Results

The experiments yielded quantitative results on algorithm performance and generated learning curve plots for visualization.

3.1. Task 1: Single-Agent Comparison

A comparative analysis based on averaged run data highlighted both performance differences between the algorithms and the impact of the environment characteristics. Across both gridworlds, a reduced exploration tendency, represented by ε=0.1, proved essential for achieving maximum path efficiency after learning stabilized. Moderate learning rates, α=0.3 or α=0.5, also tended to yield better outcomes than the slower α=0.1 setting particularly for some algorithms. In the obstacle filled Gridworld A, Double Q learning exhibited the strongest efficiency needing just 12.40 steps on average using optimal low ε settings, marginally better than Q learning's 12.63 steps and SARSA's 12.78 steps. The windy conditions of Gridworld B presented a greater challenge leading to increased step counts, where Q learning demonstrated the best adaptation finding paths averaging 14.68 steps with its best parameters, compared to Double Q learning's 15.14 and SARSA's 15.86 steps. Most parameter combinations achieving these levels of efficiency showed performance stabilization relatively early, typically within a range of 50 to 55 training episodes according to the convergence heuristic.



The learning curves displayed in Figure 1 capture SARSA's behavior in Gridworld A under nine different combinations of hyperparameters, facilitating detailed performance analysis. It highlights how stronger exploration tendencies, specifically the sets corresponding to ε=0.3 and ε=0.2, correlate with convergence to higher average step counts than seen for ε=0.1, directly reflecting how continued random actions affect overall efficiency. Examining the data for constant ε values reveals a pattern where the lowest learning rate, α=0.1, frequently leads to the highest final step averages, such as the 14.74 steps for ε=0.1 α=0.1 contrasting with 12.78 for ε=0.1 α=0.3, implying policy refinement progresses less rapidly compared to the strides made using moderate α settings like 0.3 and 0.5.

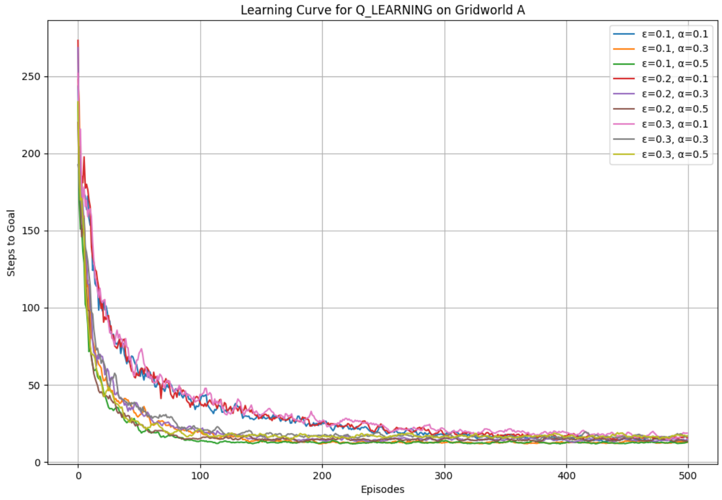
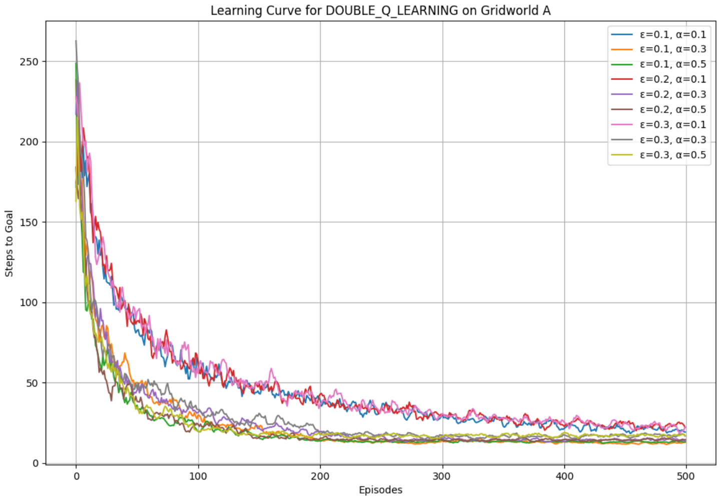


Figure 2 presents the performance of Q-learning in Gridworld A. Similar trends are visible: the curves associated with higher ε values level off at higher step counts. Again, the learning rate α=0.1 generally leads to less efficient final policies (e.g., 13.98 steps for ε=0.1, α=0.1 vs 12.63 for ε=0.1, α=0.5) compared to α=0.3 and α=0.5. For Q-learning in this environment, α=0.5 often yields the best or near-best results within each epsilon group.



The learning curves for Double Q-learning in Gridworld A are shown in Figure 3. This plot highlights a key characteristic: the significantly poorer performance when using α=0.1. The curves for α=0.1 (resulting in 20-23 average final steps) are distinctly higher than those for α=0.3 and α=0.5 (resulting in 12-17 average final steps). This suggests that a very low learning rate hinders Double Q-learning's effectiveness here. Among the better learning rates, the expected trend of higher steps with higher ε holds.

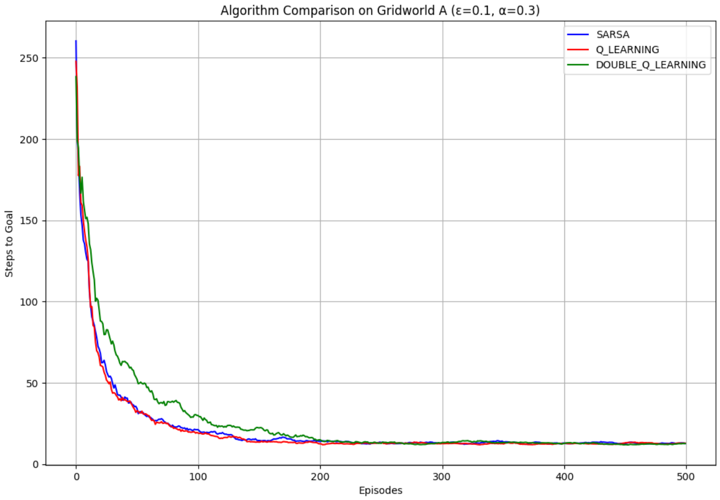


Figure 4 provides a direct comparison of the three algorithms in Gridworld A, specifically using the parameter set ε=0.1, α=0.3. For these parameters, the plot shows the Double Q-learning curve stabilizing slightly below the others (corresponding to its 12.40 step average). The SARSA curve (12.78 steps) is slightly better than the Q-learning curve (12.89 steps) in this specific instance, though all are closely clustered, indicating comparable high efficiency at these settings.

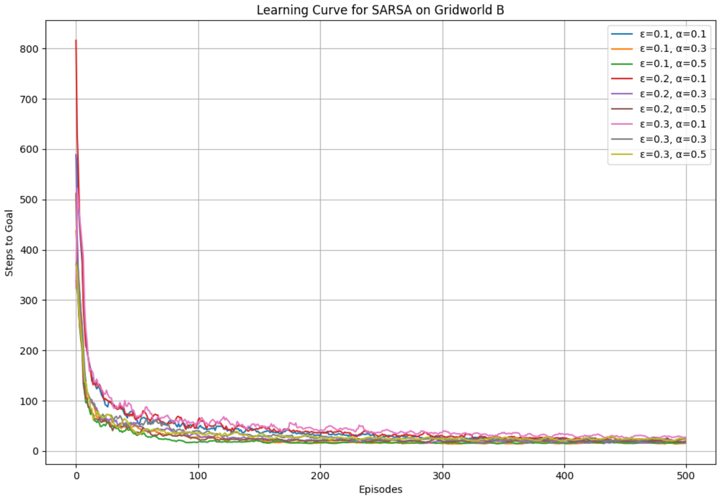
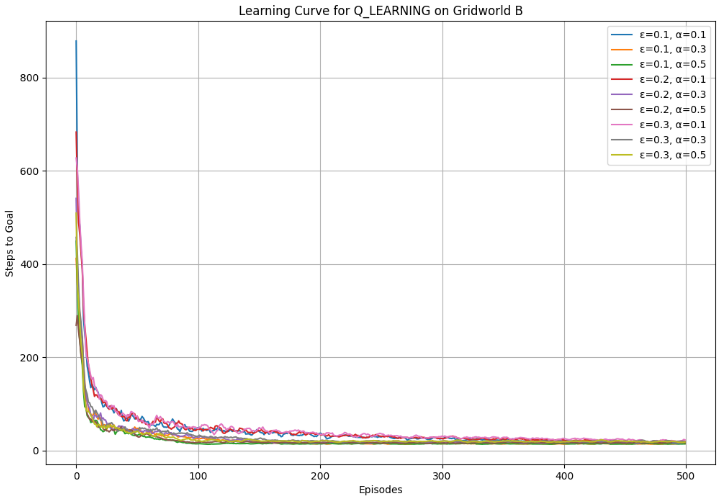
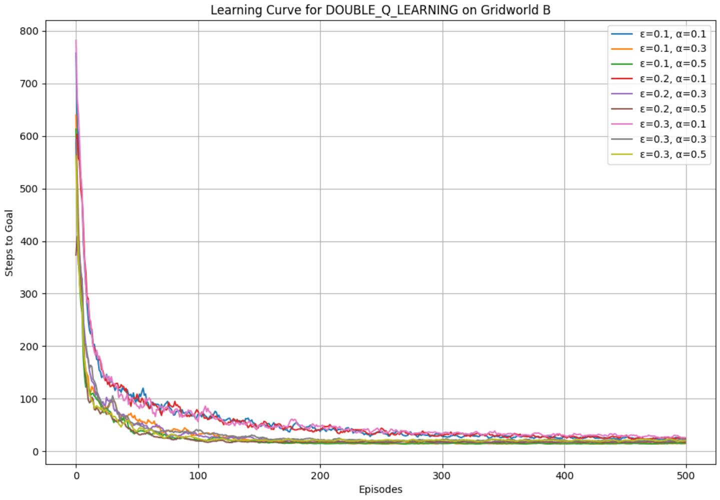


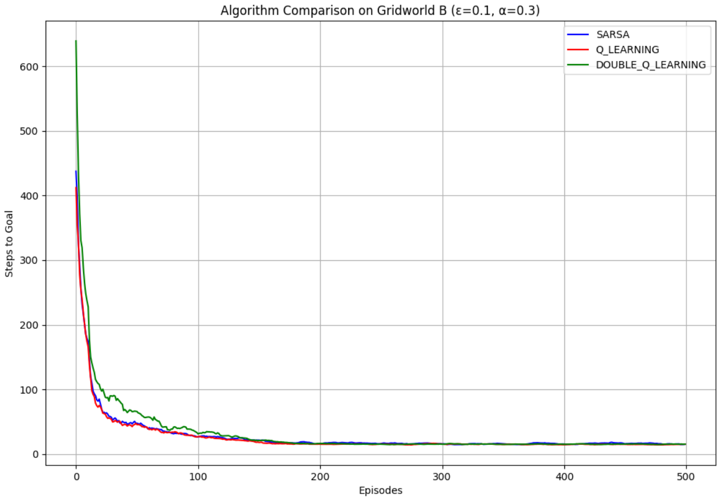
Figure 5 specifically shows how the SARSA algorithm coped with the windy conditions in Gridworld B. Comparing this visually to Figure 1 which depicted Gridworld A, one sees plainly that overall step counts are higher for every setting tested. The best average path found required 15.86 steps here versus 12.78 in Grid A, demonstrating the wind's impact. Within Figure 5, the influence of the hyperparameters ε and α follows patterns already seen. Agents exploring more frequently via higher ε values like ε=0.3 tended to take more steps per episode once stabilized. Also, the slowest learning rate α=0.1 seemed less beneficial for finding the shortest path, shown for instance by its 18.77 step result with ε=0.1 compared to the 15.86 steps achieved with α=0.3 or 16.03 steps with α=0.5.



Q learning's behavior navigating Gridworld B is visually presented in Figure 6. This algorithm managed to achieve the highest overall path efficiency recorded for this challenging windy environment. The plot shows effective learning taking place, although the general difficulty increase compared to Gridworld A seen in Figure 2 is reflected in the higher step counts the best average being 14.68 steps here contrasting with 12.63 steps before. Parameter sensitivities were consistent with previous observations as well; choosing higher values for ε diminished the efficiency of the final learned policy, and selecting α=0.1 resulted in more steps needed at the end such as 16.19 for ε=0.1 compared to 14.68 for α=0.3 or 14.72 for α=0.5 under the same exploration condition.



The performance of Double Q-learning in Gridworld B is depicted in Figure 7. Consistent with Figure 3, the curves associated with α=0.1 show markedly poor performance (24-28 final steps) compared to α=0.3 and α=0.5 (15-21 final steps). Step counts are higher overall compared to Gridworld A (Figure 3) due to the wind. While learning effectively with moderate α, its best average steps (15.14 for ε=0.1, α=0.5) were slightly higher than Q-learning's best in this environment.



Finally, the relative performance of the algorithms within Gridworld B is directly compared in Figure 8 for the parameter set defined by ε=0.1 and α=0.3. This plot visually substantiates that the Q learning approach achieved the lowest average step count after stabilization, needing 14.68 steps per episode. The corresponding curves for SARSA reaching 15.86 steps and Double Q learning achieving 15.14 steps with its best parameter setting finished slightly above this mark, lending support to the conclusion of Q learning's enhanced path efficiency when these specific hyperparameters were employed in the windy gridworld context.

3.2. Task 2: Multi-Agent Cooperative Task (IQL)

The multi agent task situated in Gridworld B where agents utilized Independent Q Learning illuminated substantial obstacles in attaining the shared goal of concurrent arrival at the target location. An analysis focusing on the progression across runs, taking into account the terminal output which tracked average steps and rewards during later training phases, revealed that successful synchronization between agents was an uncommon event. For many combinations of hyperparameters tested, notably those characterized by higher exploration factors ε=0.2 and ε=0.3 or faster learning α=0.5, it was common for agents to fail completing episodes within the 1000 step allowance. The logged performance data substantiates this observation, frequently showing average step counts stalled near the maximum limit and average total rewards remaining close to negative 3000 the value reflecting three agents accruing step costs over roughly 1000 steps throughout significant portions or the entire duration of runs conducted with parameters like ε=0.3 α=0.3 or ε=0.2 α=0.1.

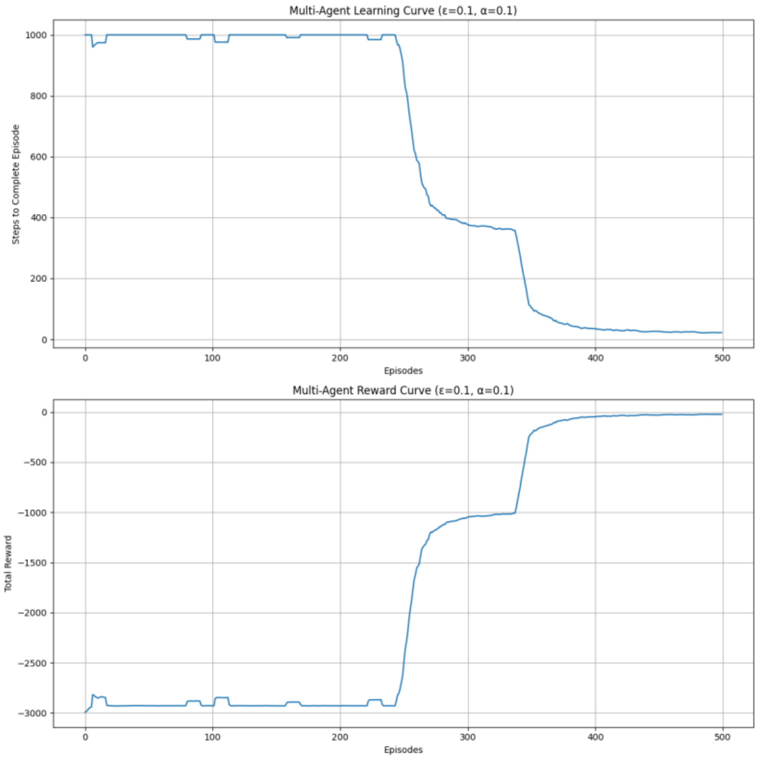
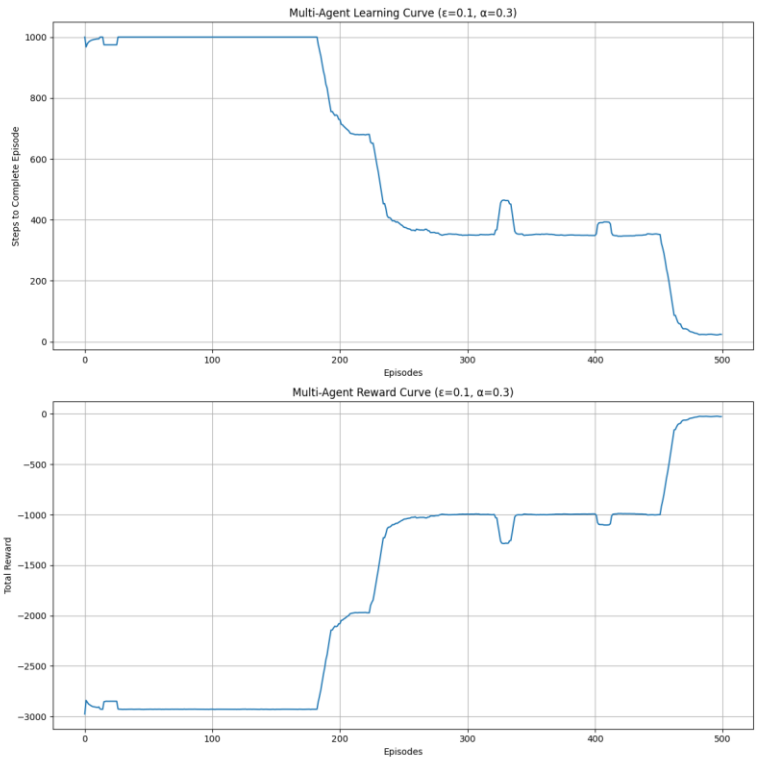
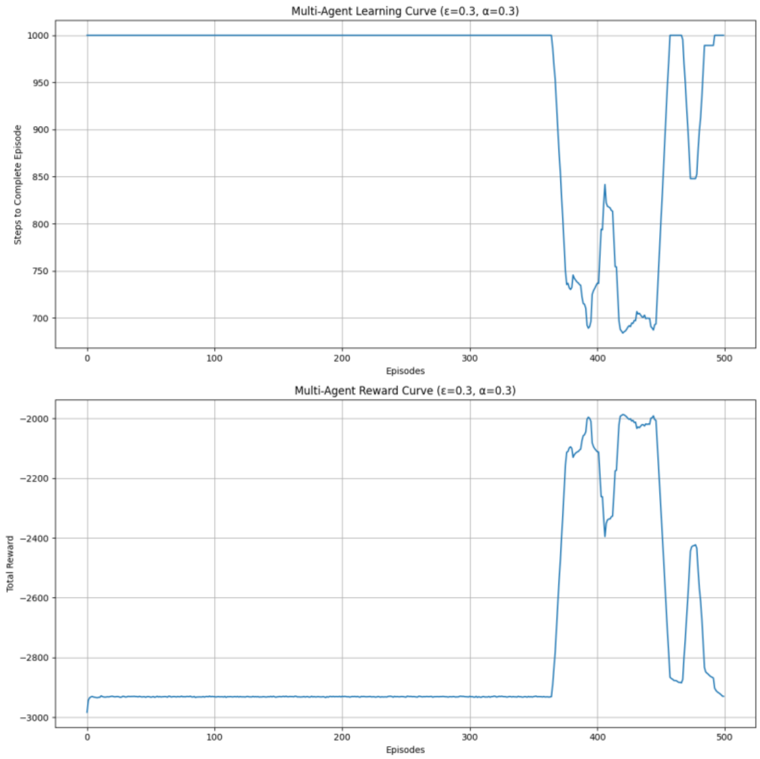


Figure 9 exemplifies one of the most successful learning outcomes observed, using low exploration (ε=0.1) and a low learning rate (α=0.1). The top plot shows that after an initial period fluctuating near the 1000-step limit, there is a distinct and relatively sharp drop in average steps starting around episode 250, stabilizing quickly at a very low value (visually well under 50 steps, consistent with terminal log averages of ~22-24 steps). Correspondingly, the bottom plot shows the average total reward mirroring this improvement, rising sharply from near -3000 to stabilize at a significantly better, but still strongly negative, level (visually estimated around -100, consistent with terminal log averages of ~ -23 to -26). This pairing demonstrates clear learning of efficient individual navigation but an evident failure to achieve the coordination needed for the positive cooperative reward.


Multi-Agent Learning Curve (ε=0.1, α=0.3)

Multi-Agent Reward Curve (ε=0.1, α=0.3)

Performance associated with the parameter combination ε=0.1 and α=0.3 is displayed in Figure 10, differing from Figure 9 only in the learning rate α. Learning commenced relatively early in these runs, noticeable from the reduction in steps starting around episode 180 to 200. The subsequent convergence however lacked the consistent smoothness seen in Figure 9. Observable in the upper plot tracking steps are clear temporary degradations, appearing as upward spikes near episodes 330 and 420, before the system returned to more efficient operation. Despite this less stable progression, the final average number of steps achieved is quite low, matching the terminal log figures showing roughly 20 to 21 steps. The total reward curve in the lower plot tracks these step count variations and stabilizes at a decidedly negative average around negative 19 to negative 21 according to terminal logs, once more signifying effective navigation learning alongside failed temporal coordination. The juxtaposition of Figure 9 and Figure 10 hints that while α=0.3 could potentially lead to slightly more efficient average individual paths, it might concurrently reduce the stability of the collective learning process compared to α=0.1.



Illustrating a common outcome when parameters were less favorable, Figure 11 showcases the results using higher exploration ε=0.3 combined with a moderate learning rate α=0.3. The upper plot demonstrates that for almost 400 episodes, the average number of steps taken remained exceptionally high, very close to the 1000 step cutoff. Following this extended period of poor performance, the system displayed highly erratic behavior, with average steps oscillating dramatically between values around 700 and 1000 and showing no sign of converging to an effective low step count strategy. The lower plot which tracks average reward follows suit, remaining stagnant near negative 3000 through most of the training before entering a period of large, corresponding fluctuations fluctuating roughly between negative 2000 and negative 2800. This figure serves as a clear graphical representation of failed learning concerning both individual navigation efficiency and group coordination, an outcome likely precipitated by the high degree of exploration disrupting convergence in the complex and constantly shifting multi agent setting.

4. Conclusions

This study successfully implemented and evaluated SARSA, Q-learning, and Double Q-learning for single-agent navigation in gridworlds featuring static obstacles (Gridworld A) and dynamic wind (Gridworld B), alongside an exploration of Independent Q-Learning (IQL) for a cooperative multi-agent task in Gridworld B.

4.1. Task 1: Single-Agent Performance Analysis

The experiments demonstrated that all three implemented algorithms: SARSA, Q-learning, and Double Q-learning, which are capable of successfully learning effective policies to navigate both Gridworld A (with obstacles) and the more dynamically complex Gridworld B (with wind). Convergence to stable, near-optimal policies was generally achieved relatively quickly, often within the first 50-60 episodes for well-tuned parameters, as indicated by the stabilization of average steps per episode.

Regarding path efficiency, a consistent if subtle advantage was observed for the off policy strategies Q learning and Double Q learning over the on policy SARSA, particularly noticeable when exploration was minimized using $\varepsilon=0.1$. Theoretically, this aligns with the fact that Q learning and its double variant directly estimate optimal values which can lead to shorter discovered paths, while SARSA's value updates are influenced by the ongoing exploration strategy. The numerical results support this interpretation; for instance, Gridworld A saw Double Q learning achieve the best average of 12.40 steps, with Q learning close behind at 12.63 steps, and SARSA slightly higher at 12.78 steps. The pattern held in the more challenging Gridworld B, where Q learning obtained the top efficiency mark of 14.68 steps, ahead of SARSA's 15.86 steps and Double Q learning's 15.14 steps.

Double Q learning demonstrated itself to be a highly competitive algorithm throughout these trials. It achieved the best overall performance within Gridworld A and maintained performance levels comparable to standard Q learning when navigating Gridworld B. Although its theoretical benefit of reducing maximization bias might manifest more clearly in environments with greater stochastic elements, the strong results obtained here serve to validate its general utility. A noteworthy aspect involved the algorithm's distinct sensitivity to the learning rate α. Employing the low setting $\alpha=0.1$ led to performance that was considerably poorer than the levels reached using either $\alpha=0.3$ or $\alpha=0.5$, underlining the necessity of an adequately high learning rate for Double Q learning to perform well.

The nature of the environment demonstrably shaped agent performance; while the static obstacles present in Gridworld A were navigated effectively after sufficient learning, Gridworld B's dynamic wind conditions introduced a persistent difficulty requiring agents to develop non intuitive compensatory movements, which consistently led to increased average path lengths for all algorithms compared to Gridworld A. Addressing these environmental challenges effectively underscored the criticality of hyperparameter tuning.

Variations in the exploration rate ε clearly illustrated the fundamental exploration exploitation trade off, with lower values like ε=0.1 proving necessary for achieving optimal final path efficiency at the potential risk of slower initial discovery, whereas higher ε values sustained less efficient paths due to ongoing random actions. The learning rate α likewise impacted outcomes, primarily governing the balance between rapid learning and stable convergence. Within this study, values of α=0.3 or α=0.5 typically provided this equilibrium more effectively than the slower α=0.1 rate, which sometimes hindered overall progress, an effect especially noticeable with the Double Q learning algorithm. Strategies involving Q learning or Double Q learning when are implemented with low exploration ε=0.1 and moderate learning rates α=0.3 or α=0.5 generally emerged as the most effective for achieving efficient paths and relatively fast stabilization, leading to robust and stable policies, as indicated by the averaged learning trajectories.

4.2. Task 2: Multi-Agent Cooperative Task (IQL)

Investigating the multi agent dynamics through Independent Q Learning IQL within Gridworld B clearly revealed the substantial challenges involved when cooperation requires accurate timing between agents. The main finding established that IQL as implemented here was substantially insufficient for enabling agents to consistently achieve simultaneous arrival at the goal, a key requirement based on the specified cooperative reward system. This limitation is a direct consequence of how IQL operates fundamentally each agent learns autonomously, effectively treating peer agents as dynamic environmental elements rather than collaborators. Such a decentralized learning strategy naturally struggles with the non-stationary environment resulting from multiple agents learning concurrently, and it incorporates no built-in mechanisms facilitating direct coordination or information exchange. Therefore, while the results showed agents could indeed learn individually effective paths under certain conditions particularly low exploration ε=0.1 coupled with low or moderate learning rates α=0.1 or α=0.3 evident from the reduced step counts in Figures 9 and 10 the crucial ability to time their arrivals together proved consistently elusive.

The failure to achieve coordinated arrival is most starkly demonstrated through the evolution of the average total rewards over time. Even when agents learned efficient individual paths, requiring only around 20 to 26 steps per episode as seen in the best cases, the average rewards achieved stayed firmly in negative territory approximately between negative 20 and negative 26, indicating a vast gap from the positive values representing successful group coordination. This outcome confirms that agent experiences were dominated by accumulating step penalties, occasionally mixed with the lesser penalty for asynchronous arrival, while the large +10 bonus for simultaneous success was seldom if ever consistently obtained. Adding to this, many different parameter settings, especially those utilizing higher exploration factors like ε=0.3 illustrated by Figure 11, prevented agents from learning even effective solo paths, leading instead to frequent episode terminations due to exceeding the step limit.

It follows clearly from these results that tackling complex cooperative objectives characterized by stringent timing requirements, such as simultaneous arrival, necessitates exploration of techniques more sophisticated than standard Independent Q Learning. Effective team strategies are difficult to form when mechanisms for information sharing, collective action planning, or direct communication between agents are absent, a fundamental limitation of the IQL approach used here.