Instituto Tecnológico de Aeronáutica - ITA
Inteligência Artificial para Robótica Móvel - CT213
Aluno: Matheus Felipe Ramos Borges

Relatório do Laboratório 11 - Aprendizado por Reforço Livre de Modelo

1 Breve Explicação em Alto Nível da Implementação

O objetivo deste laboratório é implementar um agente de Aprendizado por Reforço (Reinforcement Learning — RL) baseado em algoritmos **livres de modelo** (model-free), utilizando duas abordagens distintas: **SARSA** e **Q-Learning**. Em ambos os casos, o agente deve aprender uma política de controle que permita seguir uma pista fechada a partir da interação com o ambiente simulado.

Model-free RL significa que o agente **não tem conhecimento prévio da dinâmica do ambiente**, isto é, das funções de transição de estados ou recompensas. O aprendizado ocorre puramente com base na experiência acumulada, por meio da observação de pares (estado, ação, recompensa, próximo estado).

1.1 SARSA

O algoritmo SARSA (*State-Action-Reward-State-Action*) é uma abordagem **on-policy**, o que significa que o agente aprende com a política que está de fato sendo seguida durante a interação com o ambiente. A atualização da função ação-valor $Q(s, a)$ ocorre utilizando a **próxima ação realmente escolhida** pela política atual, normalmente com exploração via uma política ε-greedy.

A equação de atualização do SARSA é:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \right]$$

Dessa forma, a política é melhorada gradualmente com base nas ações que ela própria executa, sendo sensível ao comportamento exploratório.

1.2 Q-Learning

O Q-Learning, por outro lado, é um algoritmo **off-policy**, onde a política de aprendizado e a política de comportamento são distintas. A atualização da função $Q(s, a)$ é feita utilizando a **melhor ação possível no próximo estado**, independentemente da ação que foi realmente tomada.

A equação de atualização é:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \right]$$

Essa característica torna o Q-Learning mais agressivo na busca por uma política ótima, pois ele sempre assume que o agente tomará a melhor decisão futura possível, mesmo que não o faça na prática.

2 Figuras Comprovando Funcionamento do Código

2.1 SARSA

2.1.1 Tabela Ação-Valor e Política *Greedy* Aprendida no Teste com MDP Simples

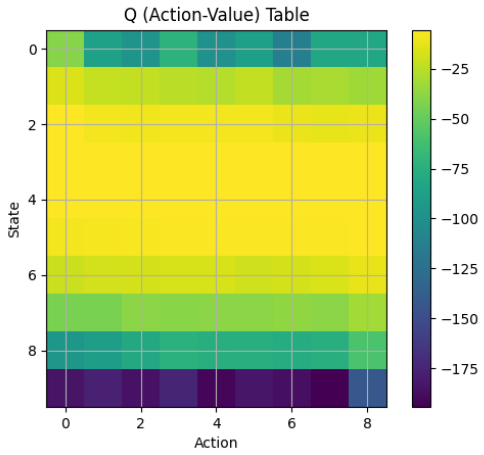

Figura 1: tabela Ação-Valor e Política *Greedy* Aprendida no Teste com MDP Simples para o SARSA.

2.1.2 Convergência do Retorno

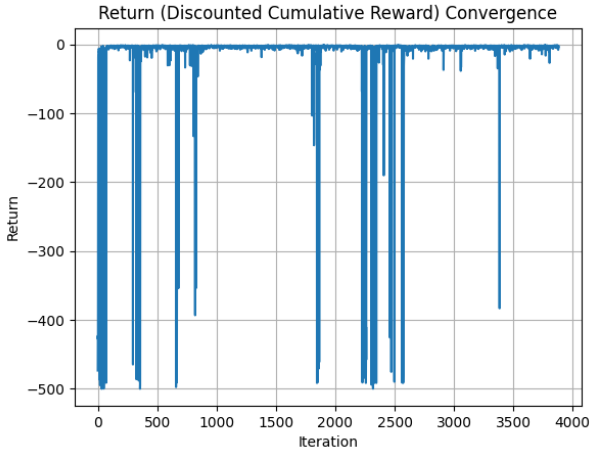

Figura 2: convergência do Retorno para o SARSA.

2.1.3 Tabela Q e Política Determinística que Seria Obtida Através de *Greedy*(Q)

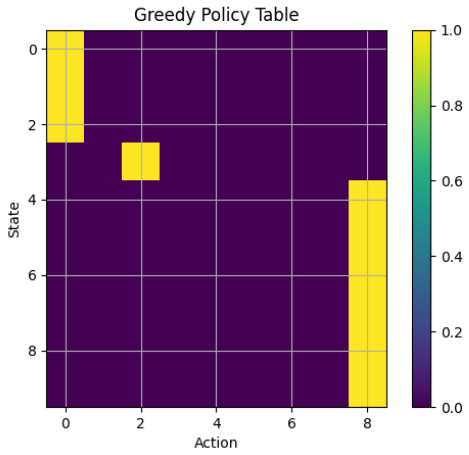

Figura 3: tabela Q e Política Determinística que Seria Obtida Através de *Greedy*(Q) para o SARSA.

2.1.4 Melhor Trajetória Obtida Durante o Aprendizado

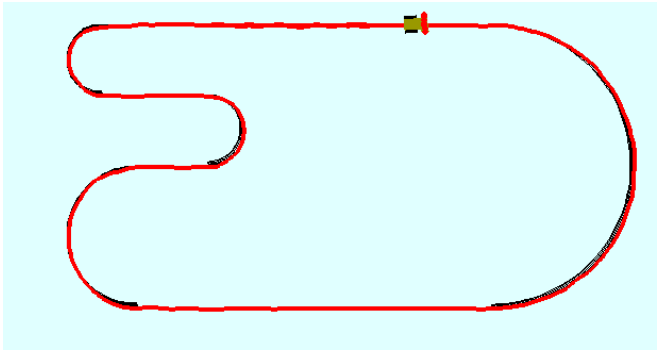

Figura 4: melhor trajetó obtida para o SARSA

2.2 Q-Learning

2.2.1 Tabela Ação-Valor e Política *Greedy* Aprendida no Teste com MDP Simples

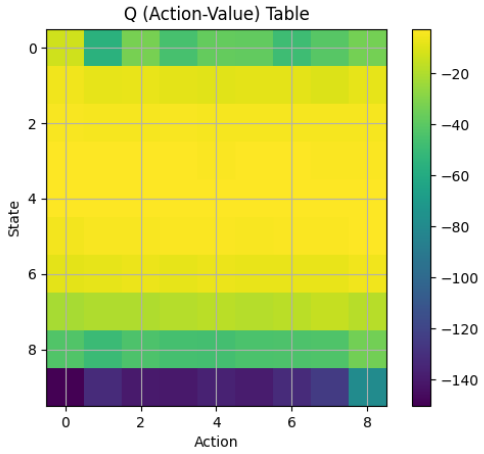

Figura 5: tabela Ação-Valor e Política *Greedy* Aprendida no Teste com MDP Simples para o Q-Learning.

2.2.2 Convergência do Retorno

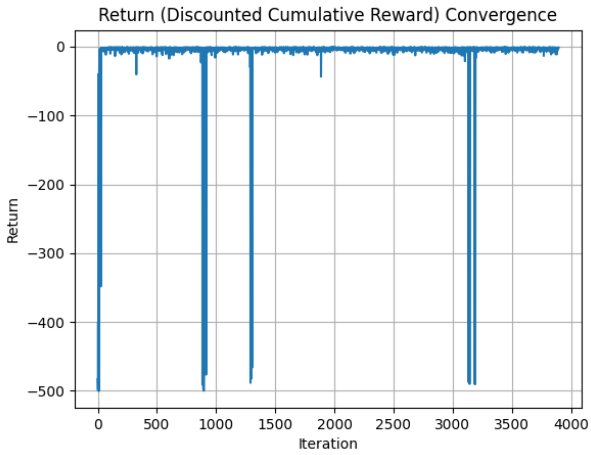

Figura 6: convergência do Retorno para o Q-Learning.

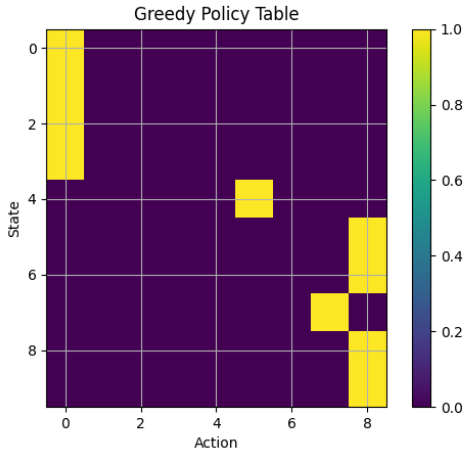

Figura 7: tabela Q e Política Determinística que Seria Obtida Através de *Greedy*(Q) para o Q-Learning.

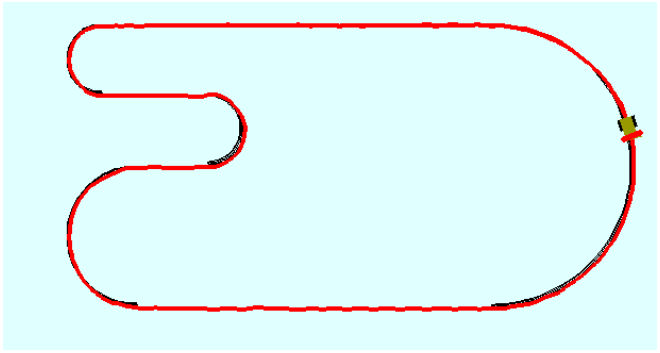

Figura 8: melhor trajetó obtida para o Q-Learning.

2.2.3 Tabela Q e Política Determinística que Seria Obtida Através de *Greedy*(Q)

2.2.4 Melhor Trajetória Obtida Durante o Aprendizado

3 Discussão dos Resultados

Os resultados obtidos demonstram que tanto o SARSA quanto o Q-Learning foram capazes de aprender políticas eficazes para seguir a pista proposta, com diferenças claras entre suas abordagens. O SARSA, por ser on-policy, apresentou convergência mais estável, porém mais lenta, refletindo o impacto direto da política exploratória no aprendizado. Já o Q-Learning, por ser off-policy, convergiu mais rapidamente para políticas mais determinísticas e eficientes, assumindo sempre a escolha da melhor ação futura. As trajetórias finais confirmam que ambos os algoritmos aprenderam comportamentos adequados, com o Q-Learning produzindo soluções mais otimizadas, enquanto o SARSA apresentou maior robustez em ambientes ruidosos ou com maior exploração.