Aprendizado por Reforço em Ambiente Grid-World para Navegação e Evasão de Obstáculos

1st Brithany Michelle Oliva Chuquimia
Matricula 971
Inatel
Santa Rita do Sapucai-MG-Brasil

2nd Luis Enrique Cardozo Ramirez
Matricula 949
Inatel
Santa Rita do Sapucai-MG-Brasil

***Abstract*—Este projeto tem como objetivo investigar o uso de Aprendizado por Reforço (Reinforcement Learning – RL) em um ambiente do tipo grid-world, com foco em problemas de navegação e evasão de obstáculos. A proposta descreve a relevância do tema, a fundamentação teórica, a hipótese e a metodologia experimental, a serem desenvolvidas no Google Colab. Espera-se que um agente treinado com algoritmos de RL, apoiados em técnicas de aproximação de função, seja capaz de aprender políticas eficazes para alcançar o objetivo com trajetórias curtas e evitando colisões. Este estudo se apoia na literatura e busca oferecer uma visão prática dos conceitos fundamentais de RL, além de estabelecer um ponto de partida para aplicações futuras em robótica e sistemas autônomos.**

***Index Terms*—Aprendizado por Reforço, Navegação de Robôs, Grid-world, Inteligência Artificial, Sistemas Autônomos.**

I. INTRODUÇÃO

O Aprendizado por Reforço (RL) é um paradigma fundamental da Inteligência Artificial (IA) no qual um agente aprende a tomar decisões sequenciais com base em interações com um ambiente dinâmico. Diferentemente do aprendizado supervisionado, em que o modelo aprende a partir de pares de entrada e saída rotulados, ou do aprendizado não supervisionado, que busca padrões em dados não rotulados, o RL opera em um cenário onde não há respostas corretas explícitas. Em vez disso, o agente é guiado por sinais de recompensa que avaliam a qualidade de suas ações em cada estado, promovendo um aprendizado orientado por tentativa e erro [1]. Esse processo permite que o agente desenvolva políticas de decisão que maximizem a recompensa acumulada ao longo do tempo, adaptando-se a ambientes incertos e dinâmicos.

Problemas de navegação e evasão de obstáculos são cenários clássicos em RL, pois encapsulam desafios práticos encontrados em aplicações reais, como robôs móveis, veículos autônomos, agentes virtuais em jogos digitais e sistemas de logística automatizada. Esses problemas exigem que o agente encontre trajetórias seguras e eficientes em ambientes com restrições espaciais, como obstáculos fixos ou dinâmicos, enquanto lida com incertezas inerentes ao ambiente. Esses cenários requerem algoritmos robustos que combinem exploração eficiente do ambiente com a capacidade de generalizar o aprendizado para novos contextos.

Antes de aplicar RL a ambientes complexos, como espaços tridimensionais ou cenários com múltiplos agentes, é comum iniciar experimentos em ambientes simplificados, como o grid-world. Esse tipo de ambiente, representado por uma matriz bidimensional, permite isolar e investigar propriedades fundamentais do aprendizado, como a convergência de políticas, o equilíbrio entre exploração e a influência da função de recompensa no comportamento do agente. O grid-world é particularmente útil por sua simplicidade computacional e capacidade de modelar problemas de navegação de forma controlada, servindo como um laboratório ideal para testar algoritmos antes de escalá-los para aplicações mais desafiadoras.

Este projeto visa investigar como um agente baseado em RL pode aprender políticas de navegação eficientes em um ambiente bidimensional com obstáculos, avaliando sua capacidade de alcançar objetivos de forma segura e com trajetórias curtas. A escolha do grid-world como ambiente experimental permite focar nos aspectos fundamentais do RL, como o design da função de recompensa e a seleção de algoritmos apropriados. Além de fortalecer a compreensão teórica dos conceitos de RL, o projeto pretende demonstrar, na prática, como esses conceitos podem ser aplicados em problemas de navegação simulada, utilizando ferramentas acessíveis como Python e Google Colab. A implementação prática será acompanhada de uma análise do desempenho do agente, considerando métricas como taxa de sucesso, número médio de passos e estabilidade do aprendizado. Este estudo serve como uma etapa inicial e fundamental para aprofundar o conhecimento necessário para futuras aplicações em robótica e sistemas autônomos, onde a navegação eficiente e segura é essencial. Além disso, o projeto busca contribuir para a comunidade acadêmica ao produzir um código didático e reprodutível, que possa ser usado como recurso educacional ou ponto de partida para extensões em ambientes mais complexos, como cenários tridimensionais ou com obstáculos dinâmicos.

II. DESCRIÇÃO TÉCNICA

O ambiente será um grid-world representado por uma matriz $N \times N$, onde cada célula corresponde a um estado. Os elementos do ambiente incluem:

- **Células livres:** regiões navegáveis.

- **Obstáculos:** células que representam barreiras; colisões resultam em penalização.
- **Meta:** célula terminal do episódio, cuja chegada resulta em recompensa positiva.

O conjunto de ações disponíveis é discreto: cima, baixo, esquerda, direita.

A função de recompensas será inicialmente definida da seguinte forma:

- Recompensa positiva ao atingir a meta: +1.
- Recompensa negativa ao colidir com obstáculo: –1.
- Recompensa neutra para movimentos normais: 0.

Se necessário, penalidades adicionais por passo (ex.: –0,01) poderão ser incluídas para incentivar trajetórias mais curtas, ajustáveis experimentalmente.

O aprendizado será realizado com algoritmos de RL, usando redes neurais para "aproximar" a função que decide qual ação o agente deve tomar. A abordagem preferencial será Actor-Critic.[5], por permitir aprender simultaneamente uma política de ações (actor) e uma função de valor (critic), sendo adequada para problemas contínuos ou de alta dimensionalidade. Como alternativa, poderá ser considerado Deep Q-Learning (DQN)[2]. A arquitetura inicial será simples (camadas densas) e poderá ser ajustada conforme os resultados obtidos durante os experimentos.

III. RELEVÂNCIA E MOTIVAÇÃO DO TÓPICO

O estudo de RL em ambientes simulados é relevante porque permite a compreensão de como agentes autônomos podem aprender comportamentos complexos a partir de regras simples, e fornece um campo de testes controlado para avaliar algoritmos que podem ser aplicados em cenários mais complexos, como robótica ou veículos inteligentes. Na prática, a navegação com evasão de obstáculos é um problema central em áreas como:

- **Robótica móvel:** robôs de entrega e aspiradores autônomos.
- **Veículos autônomos:** sistemas de direção assistida.
- **Jogos digitais e simulações:** agentes inteligentes.
- **Sistemas de logística:** otimização de rotas em armazéns automatizados.

A motivação do projeto é fornecer uma implementação prática acessível em Python/Colab, consolidando conhecimentos teóricos sobre RL.

IV. DESCRIÇÃO DO PROBLEMA

O objetivo é treinar um agente em um ambiente bidimensional com obstáculos para que ele alcance uma célula-meta de forma eficiente, minimizando o número de passos e evitando colisões. Os principais desafios incluem a definição da função de recompensas, o equilíbrio entre exploração e exploração (exploration-exploitation), e a generalização do agente para ambientes maiores.

V. HIPÓTESE

A. Hipótese principal

Um agente treinado em um ambiente grid-world é capaz de aprender políticas que minimizam o número médio de passos e maximizam a taxa de sucesso, desde que a função de recompensas seja bem definida.

B. Hipóteses secundárias

- Funções de recompensa diferentes impactam significativamente a política final aprendida.
- Estratégias de exploração com decaimento dinâmico de ϵ resultam em convergência mais rápida que valores fixos.

VI. METODOLOGIA

A metodologia será organizada em cinco fases:

1) **Implementação do ambiente:** Construção de um grid-world parametrizável em Python no Google Colab, com visualização simples para acompanhar o comportamento do agente.
2) **Modelagem da função de recompensas:** Implementação e teste de variações experimentais da função de recompensas para avaliar seus efeitos no aprendizado.
3) **Treinamento do agente:** Treinamento utilizando algoritmos de RL com redes neurais, registrando o progresso do aprendizado e ajustando parâmetros conforme necessário.
4) **Avaliação:** O desempenho será avaliado com métricas como: taxa de sucesso, número médio de passos por episódio e evolução da recompensa média.

VII. CONTRIBUIÇÕES ESPERADAS

Espera-se que este trabalho contribua para aprofundar a compreensão prática do RL, demonstrar empiricamente a influência da função de recompensas e produzir um código didático em Google Colab para fins acadêmicos e de ensino. Além disso, servirá como base para futuras extensões com ambientes mais complexos.

VIII. REFERÊNCIAS

REFERENCES

[1] R. Sutton, A. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 2ª edição, 2018.

[2] I. Grondman, L. Buşoniu, G. A. D. Lopes, R. Babuška, "A Survey of Actor-Critic Reinforcement Learning: Standard and Natural Policy Gradients", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 42, no. 6, pp. 1401–1416, 2012.

[3] V. Mnih et al., "Human-level control through deep reinforcement learning", *Nature*, 2015.

[4] S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, Pearson, 4ª edição, 2021.

[5] J. T. Springenberg et al., "Offline Actor-Critic Reinforcement Learning Scales to Large Models", *arXiv:2402.05546*, 2024.