11º Congresso de Inovação, Ciência e Tecnologia do IFSP - 2020

# Uma aplicação do método Q-learning na navegação de robôs móveis

Leonardo Di Credico Ribeiro Silva[1], Thiago Pedro Donadon Homem[2]

[1]Aluno do curso técnico integrado em Redes de Computadores, Bolsista PIBIC-EM, IFSP, Câmpus São Paulo Pirituba, leonardodicredico@gmail.com.

[2]IFSP, Câmpus São Paulo Pirituba, thiagohomem@ifsp.edu.br

Área de conhecimento (Tabela CNPq): 1.03.01.03-8 Análise de Algoritmos e Complexidade de Computação.

**RESUMO**: A navegação de robôs móveis em ambiente real é um das tarefas mais complexas em robótica, O robô deve se movimentar pelo ambiente, desviando dos obstáculos e chegar em um ponto alvo, sem qualquer intervenção humana. Este trabalho investiga e implementa o algoritmo Q-learning em simulador e em robô real. Como contribuição, a integração de ações do robô simulado com as ações do robô real permitiu que o aprendizado pudesse ser executado em simulador e replicado no robô real, evitando a excessiva repetição dos experimentos no robô real durante a fase de aprendizagem.

**PALAVRAS-CHAVE**: aprendizado por reforço; q-learning; robótica móvel.

**ABSTRACT**: The navigation of mobile robots in a real environment is one of the most complex task on robotic. The robot must move through the environment, avoiding obstacles and reaching goal, without any human intervention. This work investigates and implements the Q-learning algorithm in simulator and in a real robot. As contribution of this work, integrating the actions of the simulated robot with the real robot, allowed the learning should be executed in simulator and replicated to the real robot, avoiding the excessive repetition of the experiments in the real robot during the learning phase.

**KEYWORDS**: reinforcement learning; q-learning; mobile robotics.

## INTRODUÇÃO

A criação de robôs autônomos há tempos é um dos objetivos da robótica e um dos principais problemas desta área é o planejamento da trajetória destes robôs. O planejamento desta trajetória consiste em guiar o robô de uma posição inicial para uma posição final, buscando de preferência o trajeto mais rápido, sem colidir com os obstáculos.

Em um dos ramos da Inteligência Artificial (IA), tem-se o método de Aprendizado por Reforço (AR) (SUTTON; BARTO, 1998), um modelo de aprendizagem em que o agente interage com o ambiente e, por meio de reforços e punições, aprende a tomar decisões de maneira autônoma. Segundo (BIANCHI, 2004), no AR o objetivo do agente é aprender uma política ótima de ações que maximize a recompensa recebida durante a execução do episódio, independentemente do estado inicial onde o agente inicia no episódio. Dentre os algoritmos de AR, os mais comuns são o Q-learning e o State-Action-State-Action (Sarsa). Diversos trabalhos, ainda, investigam a utilização dos métodos de AR em robótica, como os

trabalhos de (BIANCHI et al., 2018), (HOMEM et al., 2017) e (HOMEM et al., 2020), demonstrando a aprendizagem dos agentes robóticos e a eficiência destes algoritmos.

Neste sentido, o objetivo deste trabalho é apresentar uma alternativa à tarefa de trajetória de robôs móveis em um ambiente com obstáculos, utilizando um algoritmo de AR, o Q-learning. Considerando que os métodos de Aprendizado por Reforço exigem uma quantidade considerável de repetições de episódios, grande parte dos estudos utilizam simuladores para demonstrarem a aprendizagem do agente. Realizar o aprendizado em robôs reais, além da demora considerável de aprendizagem, a repetição de ações em um ambiente real pode comprometer os sensores e dispositivos do equipamento.

Visando evitar esta situação, uma prova de conceito foi realizada integrando simulador e robô, permitindo realizar a aprendizagem inicial em um ambiente simulado e, após determinado número de iterações, as ações que são realizadas no simulador, são realizadas no robô real, numa aproximação à área de pesquisa em transferência de aprendizado.

Este trabalho está estruturado da seguinte maneira: a próxima seção apresenta a metodologia e os equipamentos utilizados, considerando a construção do robô móvel. Na sequência são apresentados os experimentos e discutidos os resultados, finalizando com as conclusões e trabalhos futuros.

## METODOLOGIA

O desenvolvimento deste trabalho iniciou-se pela revisão da literatura e estudo dos métodos de Aprendizado por Reforço. Na sequência, foram investigados os simuladores gratuitos disponíveis para serem utilizados e, por fim, foi estudada a melhor forma de construção de um robô móvel, a linguagem de programação e os dispositivos a serem utilizados no robô.

### Simuladores em robótica

Dentre os diversos simuladores gratuitos no mercado, foram selecionados dois: o Webots[1], da Cyberbotics e o CoppeliaSim[2], da Coppelia Robotics. Outro simulador, do tipo "mundo de grades", foi utilizado para aprendizagem e reutilização da aprendizagem no robô real.

O simulador Webots foi escolhido por ser um simulador 3D de robótica, gratuito, e bastante utilizado na pesquisa, ensino e indústria, além de permitir o uso das principais linguagens, como Python e C. Foi utilizada a linguagem C com os frameworks do Webots e a linguagem Python com a biblioteca third-party RPi.GPIO, com as bibliotecas built-in (sockets,time), no Raspberry para controle do robô real. Como editores de código foram utilizados o Geany, para desenvolvimento dentro do ambiente do Raspbian e o Visual Studio Code no ambiente Windows.

### Robô móvel

Para a construção do robô móvel foi utilizado como Unidade de Processamento e Controle, o Raspberry Pi3 B+. Trata-se de microcomputador do tipo Single-board, composto de um microprocessador Cortex-A53 (ARMv8) 64-bits com 1,4GHz e 1GB de memória RAM. Diferentemente dos robôs móveis mais comumente utilizados nas competições de robótica para Ensino Fundamental e Médio, que utilizam o microcontrolador Arduino, neste trabalho optou-se pelo uso do Raspberry pelo fato de permitir a execução de algoritmos de Inteligência Artificial em um sistema operacional Linux e possuir as mesmas portas de controle do Arduino (GPIO). Assim, o Raspberry é responsável por armazenar e executar os

[1] <http://www.cyberbotics.com>
[2] <http://coppeliarobotics.com>

códigos, e controlar todos os outros componentes através da GPIO. A Figura 1 apresenta o Raspberry Pi3 B+ utilizado no controle do robô móvel.

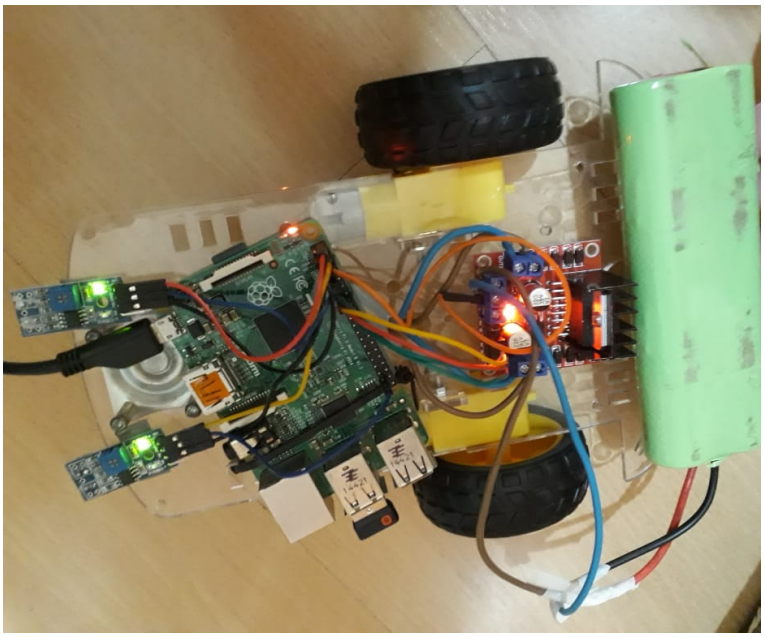

Figura 1: Protótipo do robô móvel construído pelo autor.

## Desenvolvimento

O processo de desenvolvimento do projeto deu-se nas seguintes fases: (1) Estudo dos métodos de AR; (2) Estudo de simuladores de robótica; (3) Implementação do método de AR em simulador; (4) Construção do robô móvel; (5) Controle do robô móvel; (6) Implementação do método de AR em robô móvel.

Foram estudados dois simuladores comumente utilizado em cursos de robótica, pesquisa e indústria, escolhendo o Webots como simulador de estudo. Em paralelo, foram estudados dois métodos de AR: Q-learning e Sarsa Learning e escolhido o Q-learning como sendo algoritmo de estudo. Diante da dificuldade de compreensão destes métodos, o escopo de utilização de algoritmos de AR restringiu-se no Q-learning. A partir de pesquisas, foi possível implementar o Q-learning no Webots, permitindo que, um robô móvel simulado, a partir de interações com o ambiente, após diversas iterações, aprenda o melhor entre um ponto inicial e um objetivo. Na sequência iniciou-se o processo de construção do robô móvel, com a instalação do Sistema Operacional no Raspberry Pi, os estudos para programação e controle dos motores e servos do robô. Na fase de controle do robô, além da construção das rotinas básicas de ação do robô, como andar para frente e para trás, virar à esquerda e à direita, foi construído um sistema que permite ao robô ser tele-operado. Por fim, a 6a. fase considerou a implementação do método de AR no robô real. Nesta etapa, considerando que um algoritmo de Aprendizado por Reforço exige diversas repetições para a estabilização do aprendizado, foi utilizado um simulador do tipo "Mundo de Grades", que discretiza o ambiente real em grades, criando um ambiente simulado semelhante ao ambiente real em que robô deveria aprender. As primeiras iterações são realizadas apenas no simulador e, quando verifica-se a estabilização do método, as ações realizadas no simulador são replicadas no robô, que executa a ação no ambiente real. Este experimento é detalhado na próxima seção.

## RESULTADOS E DISCUSSÃO

O experimento realizado nesta seção considerou a aprendizagem parcial em um simulador e a reutilização da aprendizagem no robô real, de modo que, a ação gerada no simulador fosse replicada no robô real. Para que fosse possível controlar um robô, de modo que as ações como andar para frente, andar para trás, andar para a esquerda e andar para a direita, não fossem ações abruptas, primeiramente foram implementados sinais de Modulação de Largura de Pulso (PWM) para que a velocidade do robô pudesse ser variada e suavizada. Dessa forma, foi possível fazer com que, as ações geradas no simulador e transmitidas para o robô fossem realizadas pelo robô mais suavemente.

O simulador do tipo "mundo de grades", que já utilizava o algoritmo Q-learning teve seu código adaptado para que as ações do algoritmo em simulador fossem enviadas ao robô. Assim, foram criadas funções que movem o robô nas 4 direções, e estas funções foram conectadas ao código de AR, de forma a fazer que o algoritmo movimentasse o agente virtual e o robô real. A Figura 2 ilustra uma ação do simulador controlando o robô real. No canto superior direito da Figura 2 é ilustrado o simulador, em que o mundo é discretizado em 5 quadrados na horizontal e 5 na vertical, o retângulo vermelho representa o robô, o objetivo é representada pelo círculo azul e os obstáculos são representados pelos triângulos. Com o objetivo de permitir a reprodutibilidade dos experimentos, os códigos desenvolvidos neste trabalho estão disponíveis <https://drive.ifsp.edu.br/s/KF2tKbY8NvD1u0P>.

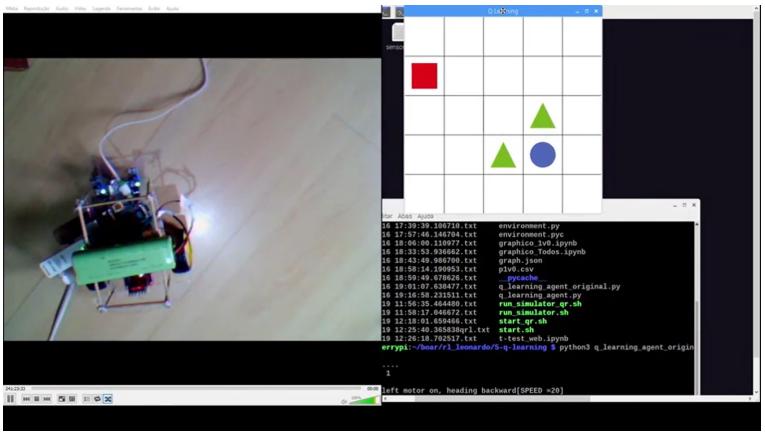

Figura 2: Exemplo da ação realizada no simulador e a replicação da ação no robô móvel.

Como forma de verificar a aprendizagem do agente executando o algoritmo de AR, conforme o ambiente ilustrado na Figura 2, foram realizados 5 experimentos com 200 episódios. A Figura 3 apresenta os resultados médios obtidos nos experimentos. É possível verificar que, a partir do episódio 40, o agente aprende o melhor caminho até o objetivo, momento em que estabiliza a aprendizagem. Assim, do episódio 1 ao 40, por exemplo, é interessante que as movimentações sejam realizadas apenas em simulador. Como verificado na Figura 3, nestes 40 episódios são necessários vários passos até o robô atingir o objetivo (círculo azul na Figura 2). A partir do episódio 40, o robô real passa a ser controlado pelo simulador e replica as mesmas ações que o simulador, atingindo o objetivo.

É importante ressaltar que, dada a simplicidade do robô móvel utilizado, verifica-se um erro acumulado de movimentação, causando, em certos experimentos, que a posição do robô no mundo real não seja a mesma esperada no simulador, mas que não invalida o trabalho desenvolvido. Tal problema pode ser facilmente corrigido por: 1) utilizar um robô com mais sensores e movimentação mais precisa ou 2) utilizar linhas que orientem o robô na movimentação. Esta segunda proposta é apresentada como trabalhos futuros nas conclusões.

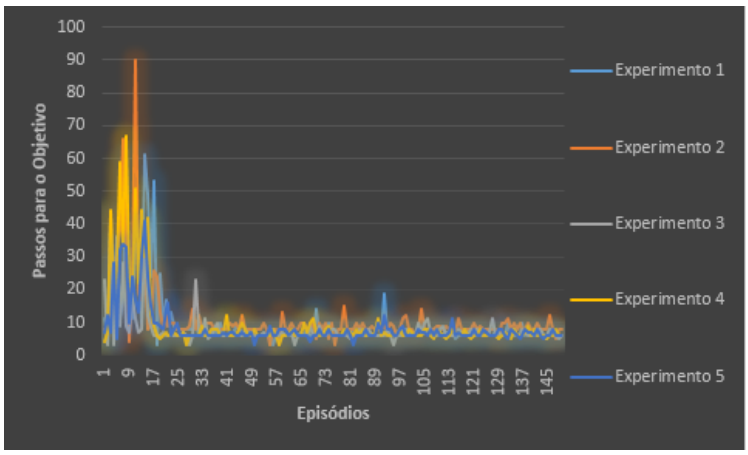


Figura 3: Gráfico da média dos passos por episódios até o objetivo.

## CONCLUSÕES

Este trabalho implementou o algortimo Q-learning em simulador e em robô real, permitindo que o agente aprenda uma trajetória ótima em ambientes com obstáculos. A principal dificuldade foi verificada nos experimentos em ambiente real, com problemas de conexões ou calibragem nos movimentos, que exigiam reiniciar os experimentos. Dada a necessidade de inúmeras repetições para convergência do algoritmo, como contribuição deste trabalho utilizou-se uma estratégia que se assemelha à reutilização da aprendizagem. Utilizando-se um simulador simples, que representa o mundo discretizado em grades, a aprendizagem iniciava-se neste ambiente e, após algumas dezenas de iterações, a ação que o agente deveria realizar no simulador era replicada ao robô real. Desta forma, toda a fase de aprendizado, que exige consumo de bateria e desgaste dos motores, foi substituída e acelerada no simulador. Como proposta futura, propõe-se a utilização de linhas em forma de grade, que serviriam de guias para os robôs móveis se deslocarem dentro de um ambiente e corrigir qualquer variação gerada durante a movimentação do robô.

## AGRADECIMENTOS

Leonardo agradece o apoio da CNPq, através da concessão de bolsa de PIBIC-EM. Thiago agradece o apoio da PRP/IFSP e da direção do Câmpus PTB/IFSP no desenvolvimento deste trabalho.

## REFERÊNCIAS

BIANCHI, R. A. C. *Uso de heurísticas para a aceleração do aprendizado por reforço*. 174 f. Tese (Doutorado em Engenharia Elétrica) — Universidade de São Paulo, São Paulo, 2004.

BIANCHI, R. A. C. et al. Heuristically accelerated reinforcement learning by means of case-based reasoning and transfer learning. *Journal of Intelligent & Robotic Systems*, v. 91, n. 2, p. 301–312, Aug 2018. ISSN 1573-0409.

HOMEM, T. et al. Qualitative case-based reasoning and learning. *ARTIFICIAL INTELLIGENCE REVIEW*, v. 283, jun. 2020. ISSN 0269-2821.

HOMEM, T. P. D. et al. Improving Reinforcement Learning Results with Qualitative Spatial Representation. In: *2017 Brazilian Conference on Intelligent Systems (BRACIS)*. [S.l.]: IEEE, 2017. p. 151–156. ISBN 978-1-5386-2407-4.

SUTTON, R. S.; BARTO, A. G. *Reinforcement Learning : An Introduction*. [S.l.]: MIT Press, 1998.