

# Navegação e Prevenção de Colisões de um Robô de Rodas Diferencial utilizando Q-learning

Hugo Tallys

Instituto de Computação  
Universidade Federal de Alagoas

COMP268  
Inteligência Artificial 2 2021.2



# Sumário

- 1 Introdução
- 2 Modelagem do problema de AR
- 3 O algoritmo
- 4 Resultados



1 Introdução

2 Modelagem do problema de AR

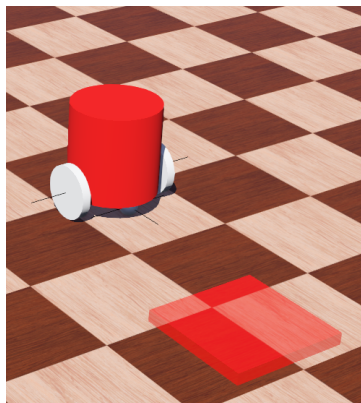
3 O algoritmo

4 Resultados



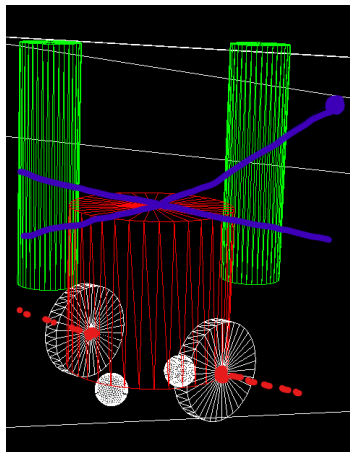
# Definindo o problema

- Controlar um robô de rodas diferencial
- Navegação (objetivo) e prevenção de colisões (obstáculos)



# Definindo o problema

- Controlar um robô de rodas diferencial
- Navegação (objetivo) e prevenção de colisões (obstáculos)
- Sensores e atuadores



1 Introdução

2 Modelagem do problema de AR

3 O algoritmo

4 Resultados



## Estado do agente

Estado do agente no instante  $t$ :

$$s_t = (L_1, L_2, L_3, L_4, d, \theta)$$

Onde:

- $L_i$  = leitura do sensor lidar  $i$
- $d$  = distância ao objetivo
- $\theta$  = ângulo ao objetivo (coordenada polar)



## Ações do agente

As ações são definidas da seguinte maneira:

$$a_i, i \in 0, 1, 2$$

Onde:

- $a_0 \rightarrow (v, \omega) = (0.05, 0)$
- $a_1 \rightarrow (v, \omega) = (0, \pi/2)$
- $a_2 \rightarrow (v, \omega) = (0, -\pi/2)$

Obs.:  $\Delta t = 1s$  (intervalo de atraso) onde o controlador ficara ocioso sem tomar nenhuma ação.





## Recompensas

As recompensas dadas ao agente são definidas da seguinte maneira: Onde:

- $r_t = +5.000$  se o agente chega ao objetivo
- $r_t = -5.000$  se o agente colide com algum obstáculo
- $r_t = \cos(\phi)$  caso contrário

Obs.:  $\phi$  é o ângulo entre a direção do objetivo e a direção que o robô avança.

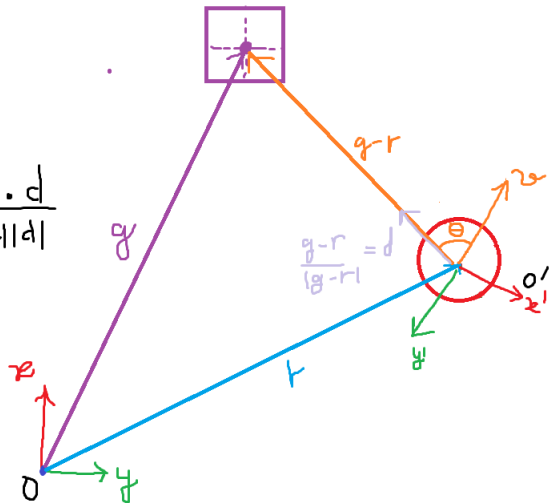


# Modelagem III

$$R : D \rightarrow O'$$

$$v = R \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\cos \theta = \frac{v \cdot d}{|v||d|}$$



1 Introdução

2 Modelagem do problema de AR

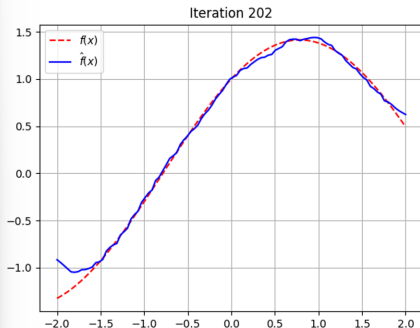
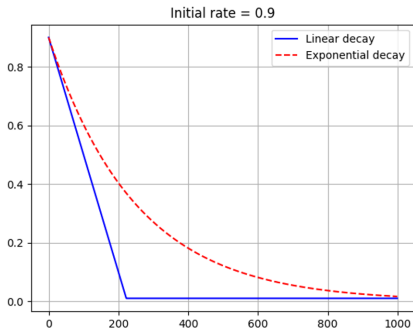
3 O algoritmo

4 Resultados



# Modelagem III

- Tile coding (CMAC) (64, 128 Tilings  $2 \times 2 \times \dots$ ,  $5 \times 5 \times \dots$ ,  $10 \times 10 \times \dots$ )
- Política  $\epsilon$ -greedy com taxa de exploração decaindo exponencialmente  $\epsilon(k) = 0.9(0.996)^k$ .



- 1 Introdução
- 2 Modelagem do problema de AR
- 3 O algoritmo
- 4 Resultados**

