

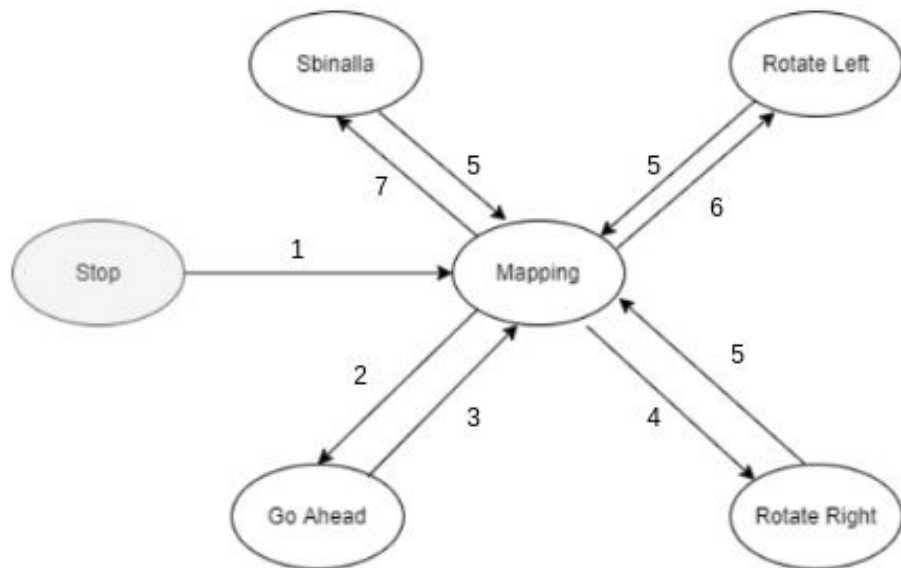
universidade de aveiro
theoria poiesis praxis

Robótica Móvel e Inteligente


Dário Matos - 89288
Pedro Almeida - 89205

Máquina de estados

1. Início
2. Decisão de andar em frente
3. Posição objetivo alcançada
4. Decisão de rodar -90°
5. Orientação objetivo alcançada
6. Decisão de rodar $+90^\circ$
7. Inversão de sentido (rodar 180°)



Movimento linear

- Mantêm a orientação
- Limiar de distância lateral a obstáculos 
- Verificar se chegou ao objetivo

$$\text{rot} = k * (m - r)$$

$$\text{r_power} = \text{lin} - (\text{rot} / 2)$$

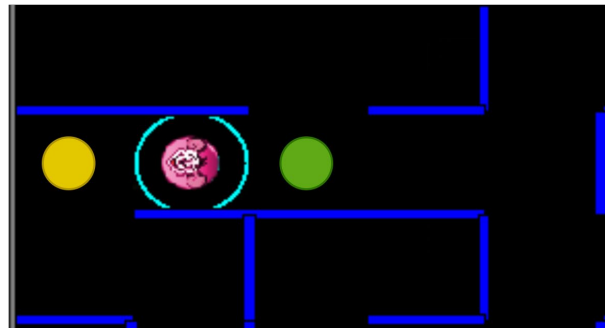
$$\text{l_power} = \text{lin} + (\text{rot} / 2)$$

Movimento de rotação

- Cálculo da orientação objetivo
 - Casos especiais para cada estado
- Aplica-se uma potência inversa em cada roda
- Reduz-se a potência aplicada quando a diferença da orientação atual e a orientação objetivo é menor que 30°
- Verificar se alcançou orientação objetivo (intervalo de 5° para cada lado)

Cálculo da próxima posição objetivo

- Com os sensores de proximidade verifica o seu redor:
 - Escolhe uma das posições livres
 - Guarda as restantes posições livres que não foram selecionadas
- Se não tiver nenhuma posição livre em seu redor:
 - Algoritmo A* para a posição não visitada mais perto
- Posição objetivo é mantida até ser alcançada
- Se tiver um caminho definido, objetivo é a próxima célula do caminho



Modelo de movimento - cálculo da localização

$$out_t = \frac{in_i + out_{t-1}}{2} * N(1, \sigma^2)$$

$$x_t = x_{t-1} + lin * \cos(\theta_{t-1})$$

$$y_t = y_{t-1} + lin * \sin(\theta_{t-1})$$

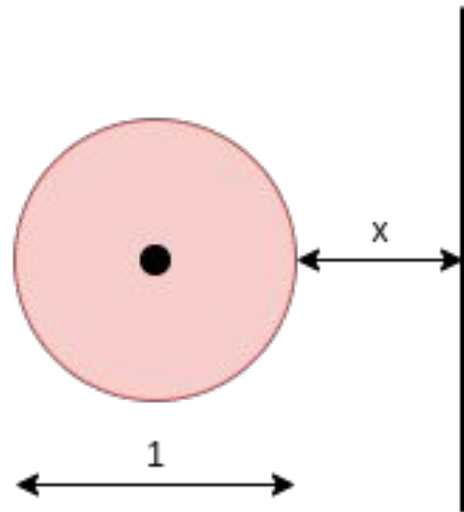
$$lin = \frac{out_t^l + out_t^r}{2}$$

- Comparado com sensor GPS, o GPS teórico tem tendência a estar adiantado.
- Problemático uma vez que o erro vai acumulando

Correção da localização

- Redução da variável *lin* quando última potência enviada para os motores foi zero.
- Correção feita com a posição das paredes:

Posição = $p_{\text{parede}} - \text{espessura} - x - \text{raio_robo}$



Mapeamento

Objetivo: fazer mapeamento do mapa desconhecido

- Guarda num dicionário onde a chave é a célula do mapa e o valor é se a posição está livre ou não. Exemplo $\{(2,0): "X"\}$
- Sempre que há informação nova, escreve-se o ficheiro de *output*

Planeamento

Objetivo: Calcular o circuito fechado de menor custo que passe por todos os *spots*. (Problema do caixeiro viajante)

- Durante a exploração do mapa é guardada a posição dos *spots*
- Construção de um grafo com o custo do caminho de cada *spot* a todos os outros
- Algoritmo que permite resolver o problema do caixeiro viajante (calcula todas as permutações)
- Escrita do resultado no ficheiro de *output*