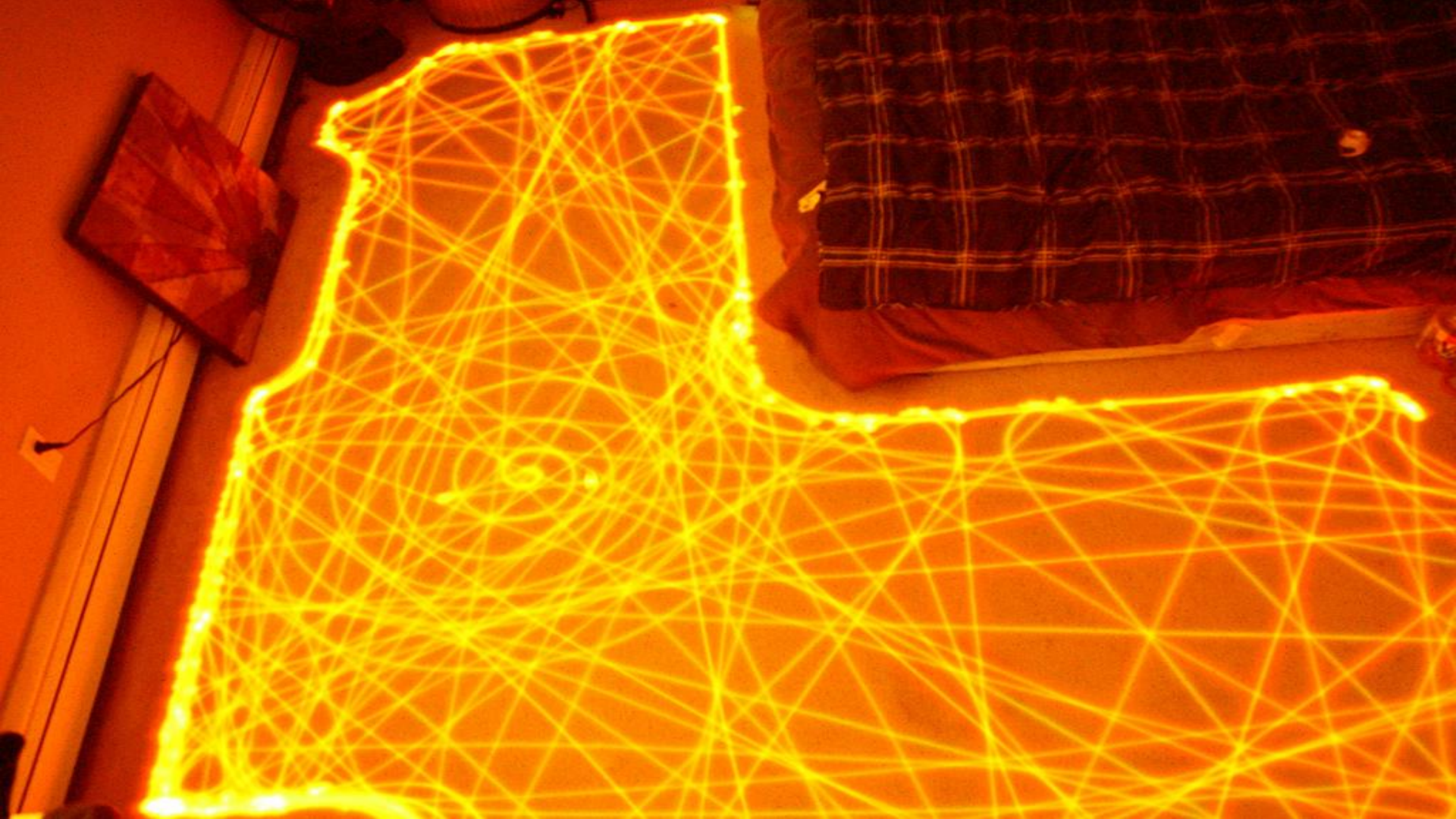


Robótica Computacional

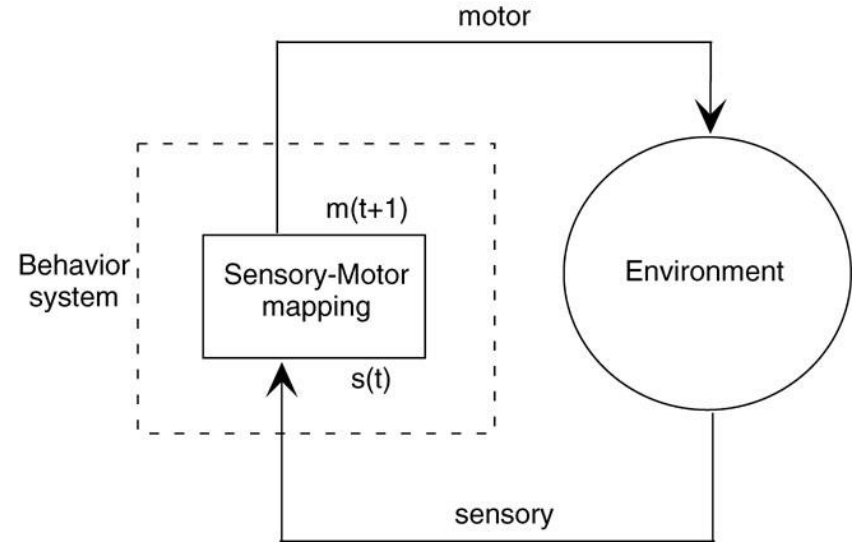
Introdução a ROS





Estrutura de controle de robô

- Um robô autônomo vê o ambiente ao redor e depois executa uma ação.
- Dependendo do seu comportamento, podemos ter diferentes resultados.



Exemplos de Robôs

UVD Robot

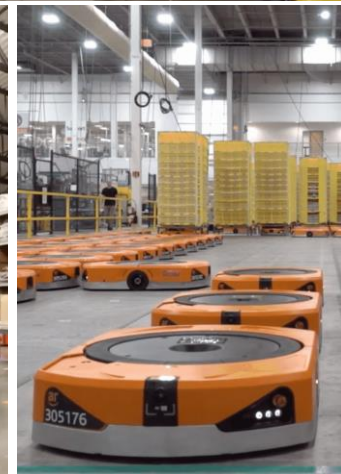
- Luz ultravioleta
- Câmera 3D
 - Localização
- Sensores de profundidade
 - Detector de obstáculos
- Comunicação wireless
- Inteligência
 - Precisa saber onde o robô já visitou
 - Evitar pessoas!



Exemplos de Robôs

Amazon Robotics fulfillment center

- Robôs
 - Fanuc M-2000iA
 - Robotic Drive Unit
- Sensores
 - Lidar (2D / 3D)
 - Câmera RGB
 - Distância (Ultrasom ou IR)
- Inteligência
 - Deve entregar a “pod” do ponto A até o ponto B
 - Evitar muitos outros robôs



Mas e se o ambiente não for estruturado?



Competição da DARPA Subterranean Challenge

Objetivo

- Navegar em um ambiente perigoso
- Encontrar objetos de interesse
 - Resgate
 - Gás
 - Artefatos

Como vencer uma competição dessas?

- Gerar mapas do ambiente
- Manter localização do robô
- Detectar objetos de interesse e mapear sua localização
- Coordenar diversos tipos de robô



Vencedor: Team Cerberus

Robôs:

- Spot (payload)
 - four-legged
 - 360o camera
 - LIDAR
 - Microphone
- Seekur Jr: (mapeamento)
 - six-whelled
 - LIDAR
 - GPS
- DroneSense (exploração)
 - 360o camera
 - LIDAR
 - Detector de gás
- Flyability Gimball (exploração)
 - 360o camera
 - LIDAR

Como administrar essas informações?



ROS - Robot Operating System

“open-source robotics middleware platform that provides a set of software libraries and tools for building robot applications.”

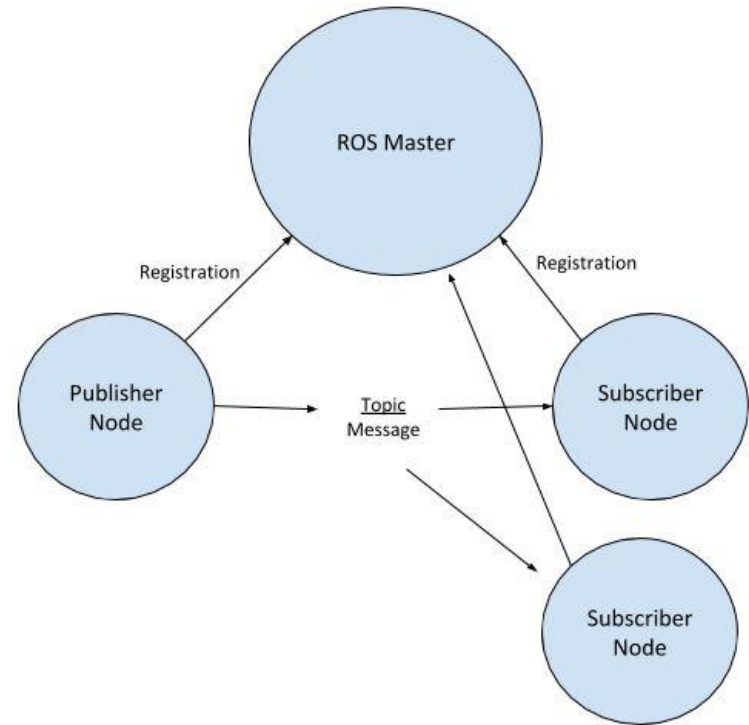
- Vantagens da ROS
 - Modular
 - Sistema de mensagem
 - Visualizatização
- Versão: Noetic



ROS - Robot Operating System

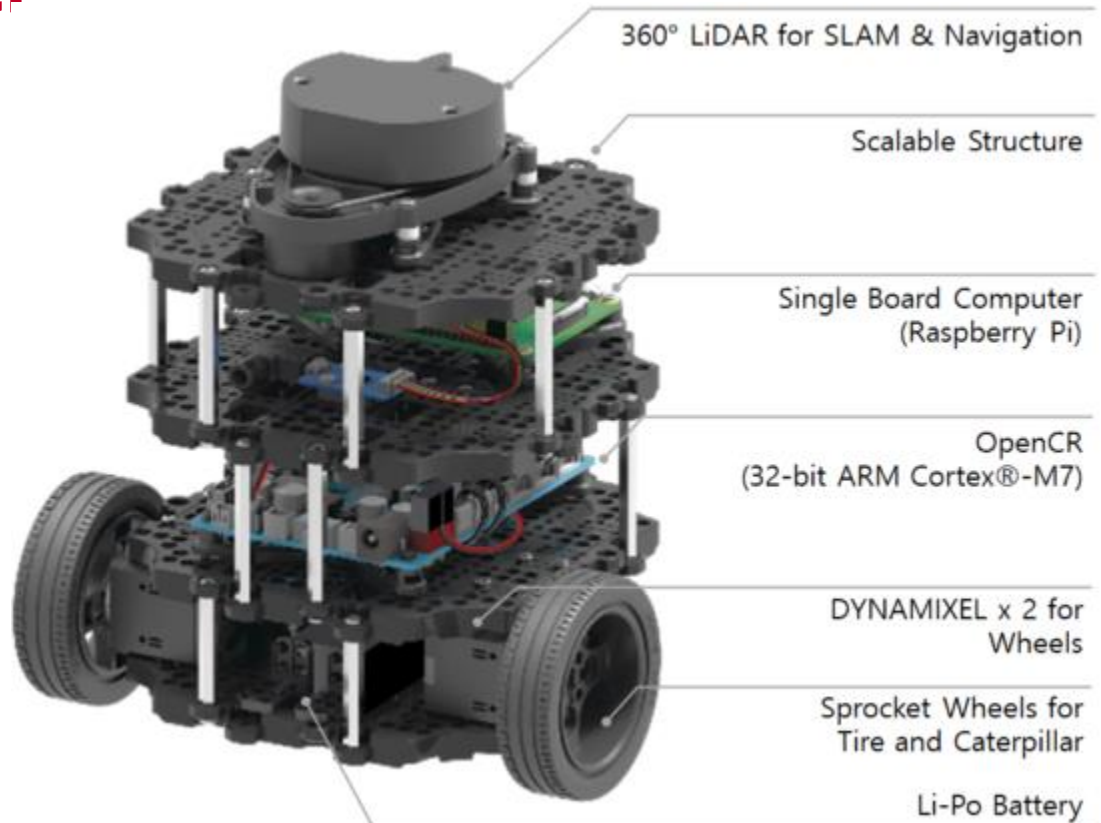
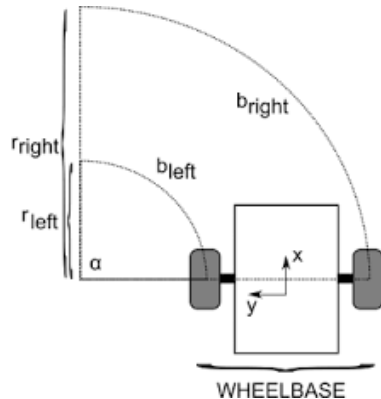
Sistema da ROS

- ROS Master: Administrador
- Node: Script do python
- Topic: Barramento onde trocamos informações entre nós.
- Message: Formato de dados
- Publisher: Nó que publica a mensagem.
- Subscriber: Nó que recebe a mensagem.



Turtlebot Burguer

- Sensores
 - Wheels Encoder
 - LiDAR
 - Camera RGB
- Odometry
 - 1 rotation = $2\pi R$
 - $\Delta\theta = \omega t + \frac{1}{2}\alpha t^2$



Localização

- LiDAR (Sensor Laser)
 - Max: 3.5 m
 - Length: 360
 - 1 deg / step
- Com base na leitura do sensor o robo consegue estimar a sua posição.

