

# SISTEMA DE NAVEGAÇÃO PARA ROBÔS MÓVEIS BASEADO EM SLAM USANDO SENSORES DE BAIXO CUSTO

---

PEDRO HENRIQUE PAIOLA RA 161020305

ORIENTADOR: PROF. DR. HUMBERTO FERASOLI FILHO

# Introdução

---

- Navegação de robôs móveis.
- Obstáculos.
- Características do ambiente.
- Abordagens.
- Sistema de mapeamento e navegação.

# Justificativa

---

- Diversas aplicações:
  - Transporte de materiais;
  - Veículos autônomos;
  - Exploração de ambientes inóspitos e de difícil acesso.
- Aplicação de conhecimento obtido durante o curso.
- Sistema de mapeamento e navegação com sensores de baixo custo.

# Objetivos

---

- Implementar um sistema de navegação de robôs, usando técnicas de SLAM (*Simultaneous Localization and Mapping*).
  - Programar funções básicas do robô;
  - Implementar a comunicação entre o robô e o computador;
  - Tratar dados dos sensores ultrassônico e infravermelho;
  - Implementar a solução de mapeamento do ambiente;
  - Implementar algoritmos para navegação do robô;
  - Realizar testes e avaliar os resultados.

# Metodologia

---

- Materiais utilizados: robô Frank e computador do autor.
- Linguagens: C++ e Java.
- Etapas:
  1. Levantamento bibliográfico;
  2. Definição da solução;
  3. Implementação da solução e testes iniciais;
  4. Testes finais e avaliação dos resultados.

# Navegação de Robôs Móveis

---

- Matarić (2007) define um robô como “um sistema autônomo que existe em um mundo físico, pode perceber o ambiente a sua volta e pode atuar nele para alcançar seus objetivos”.
- Problema da navegação de robôs móveis: robôs capazes de se locomover pelo ambiente, cujo o objetivo é obter sucesso no seu deslocamento de um ponto inicial a um ponto objetivo, livre de colisões.
- Diversas dificuldades são encontradas neste processo.






# Navegação de Robôs Móveis

---

- Principais problemas que podem estar envolvidos:
  - Localização;
  - Busca;
  - Planejamento de trajetória;
  - Cobertura;
  - SLAM (*Simultaneous Localization and Mapping*).
- Abordagens empregadas:
  - Deliberativa;
  - Reativa;
  - Híbrida.

# Abordagem deliberativa

---

- O robô necessita de um modelo prévio do ambiente: um mapa
  - Ambiente estático e conhecido.
- Problemas enfrentados:
  - Localização 
  - Busca 
  - Planejamento de trajetória 
  - Cobertura 
  - SLAM 
- Foco das soluções: planejar trajetórias.
- Classes de métodos: mapeamento de caminhos, decomposição em células e método dos campos potenciais.



# Abordagem reativa

---

- Sem conhecimento prévio do ambiente, o robô toma decisões baseando-se nas leituras atuais de seus sensores.
- Estímulo => reação.
- Permite lidar com ambientes complexos, desconhecidos e dinâmicos.
- Não adquire conhecimento sobre o ambiente.

# Abordagem híbrida

---

- Visa unir as vantagens das abordagens anteriores e minimizar suas desvantagens.
- Estratégias de controle baseadas em comportamento.
- Exemplo para navegação:
  - Mapeamento do ambiente antes, ou durante a navegação.
  - Planejamento de trajetória utilizando o mapa, porém desviando de obstáculos detectados pelos sensores que não estavam presentes no mapa.

# Localização

---

- Problema da localização: determinar a posição do robô no ambiente em que ele está inserido.
- “Onde estou?”
- Problema fundamental para a navegação e para o mapeamento.
- Categorias de métodos: Localização Relativa, Localização Absoluta e Fusão de Multi-sensores

# Localização Relativa

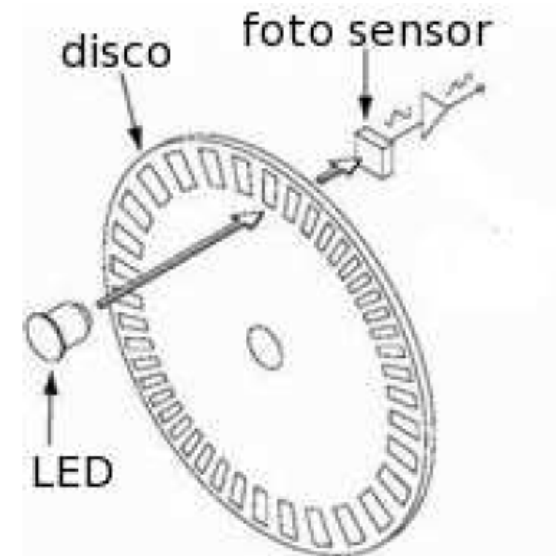
---

- Localização obtida a partir das estimativas de localizações anteriores, unidas com informações como:
  - Comandos de movimentação;
  - Dados obtidos pelos sensores.
- Desvantagem: propagação e acúmulo de erros.

# Localização Relativa

---

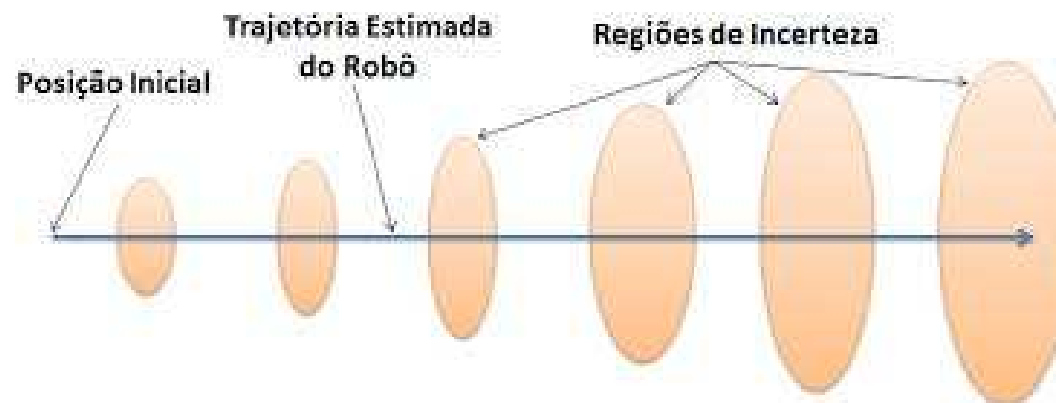
- Odometria: determinação da localização do robô a partir do movimento das rodas.
  - Encoders: sensores de rotação que contam a quantidade de giros de cada roda.
  - Baixo custo.
  - Acúmulo de erros.



# Localização Relativa

---

- A odometria está sujeita a dois tipos de erros:
  - Erros sistemáticos: gerados devido às incertezas nos parâmetros do modelo cinemático do robô.
  - Erros não sistemáticos: devido a situações inesperadas.



# Localização Absoluta

---

- Localização obtida independentemente de localizações previamente calculadas.
- A localização atual não é derivada de integrações.
- Sem acúmulo de erros.
- Três métodos principais: utilizando *beacons*, *landmarks* ou baseado em mapas.

# Localização Absoluta

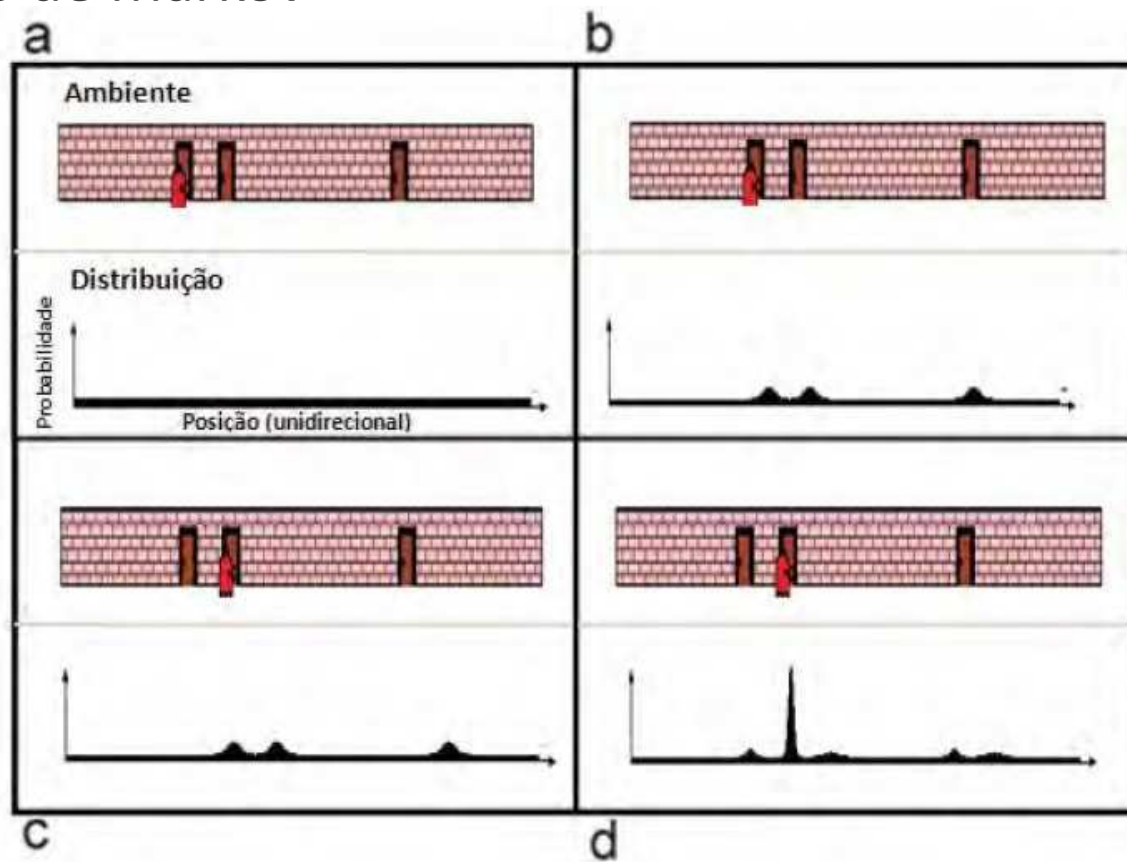
---

- Baseado em mapas: com um mapa do ambiente disponível, medições sensoriais sobre o ambiente são adquiridas e comparadas (*matching*) com as informações do mapa.
  - Alto custo computacional
  - Necessidade de sensores precisos



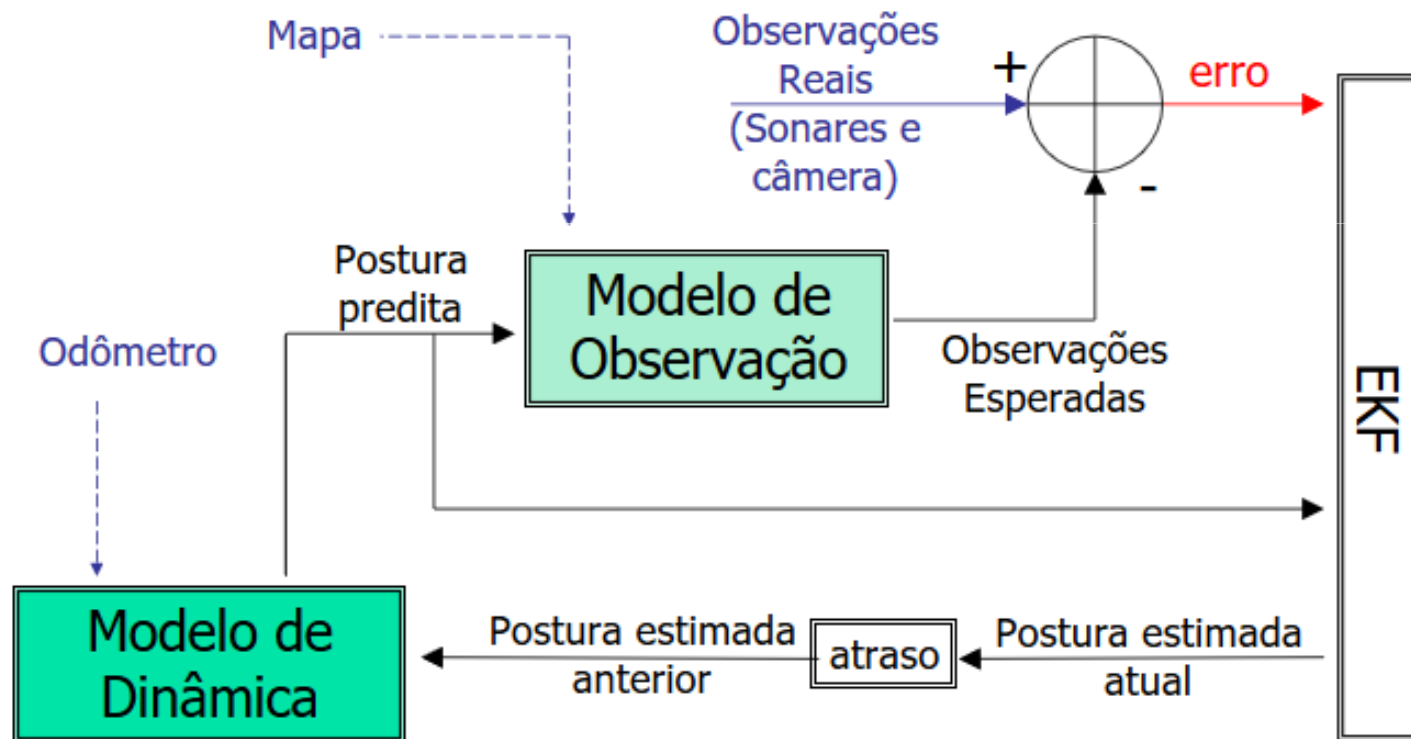
# Localização Absoluta

- Método de Markov



# Localização Absoluta

- Filtro de Kalman



# Localização Absoluta

---

- Filtro de Partículas
  - Também conhecido como método de Monte-Carlo.
  - Filtro baseado em amostras para redes Bayesianas dinâmicas.
  - Trata qualquer tipo de distribuição de probabilidade, diferente do Filtro de Kalman.
  - Pode ser usado tanto para localização global como local.

# Localização Absoluta

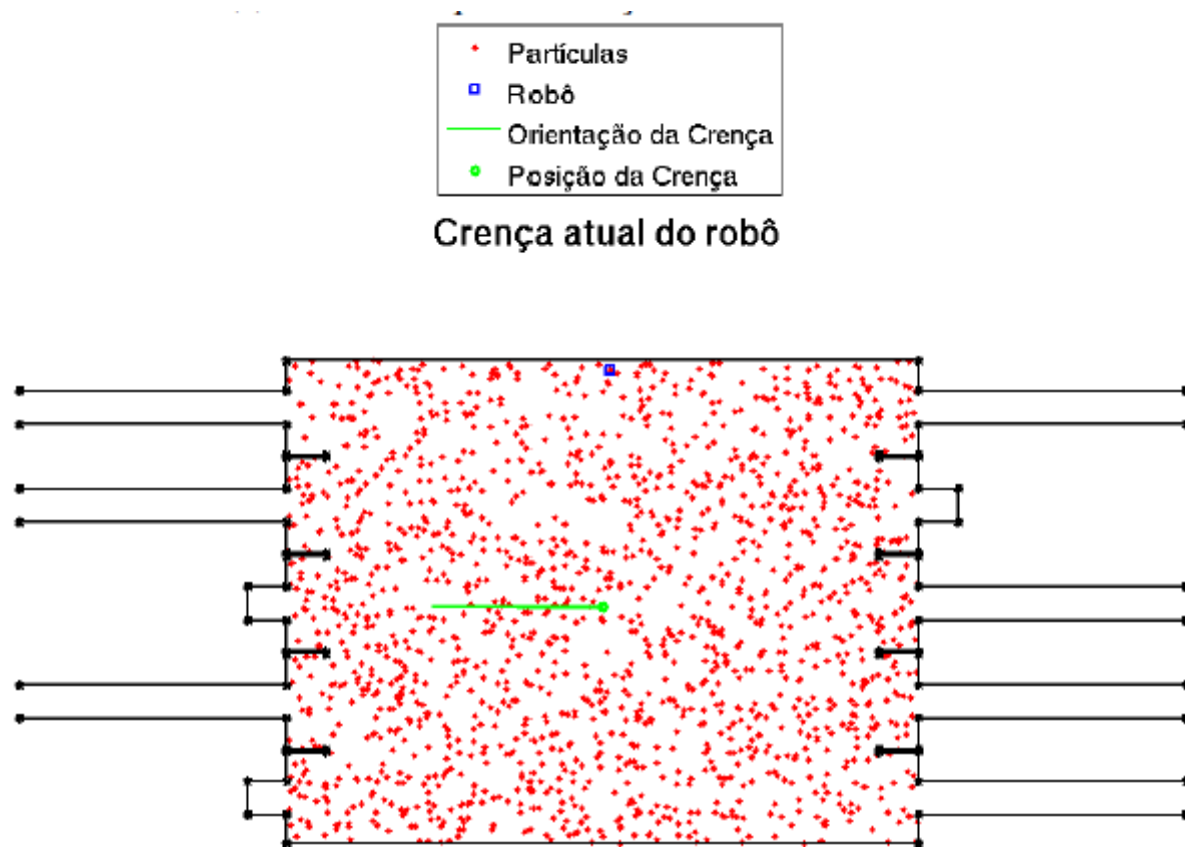
---

- Filtro de Partículas

- Utilização de um conjunto de partículas, onde cada partícula representa uma possível posição  $(x, y, \theta)$  do robô.
- A partir das observações do robô, determina-se a “crença” de cada partícula, a probabilidade do robô estar na posição referente aquela partícula.

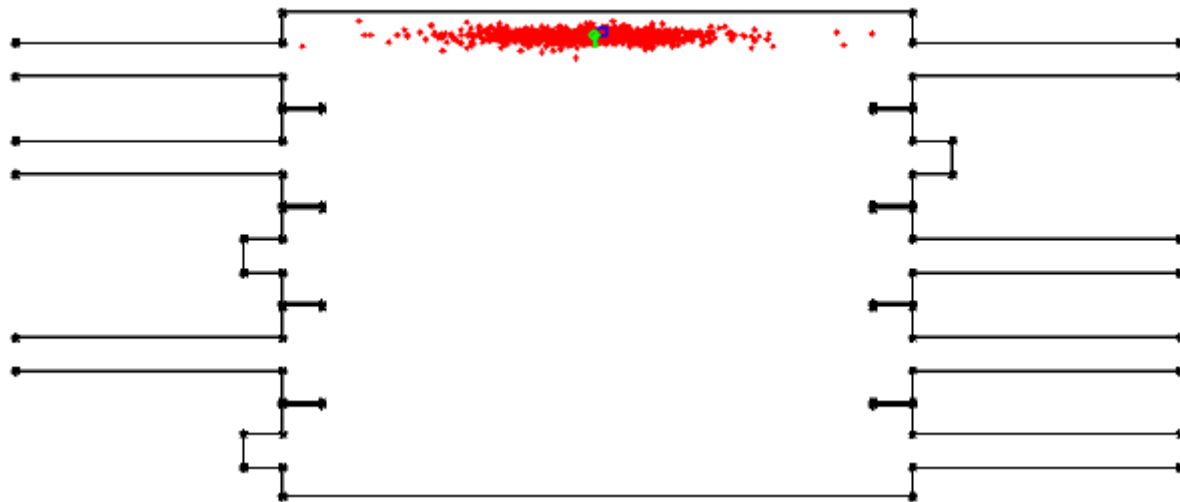
# Localização Absoluta

---



# Localização Absoluta

---



# Localização Absoluta

---



# Mapeamento

---

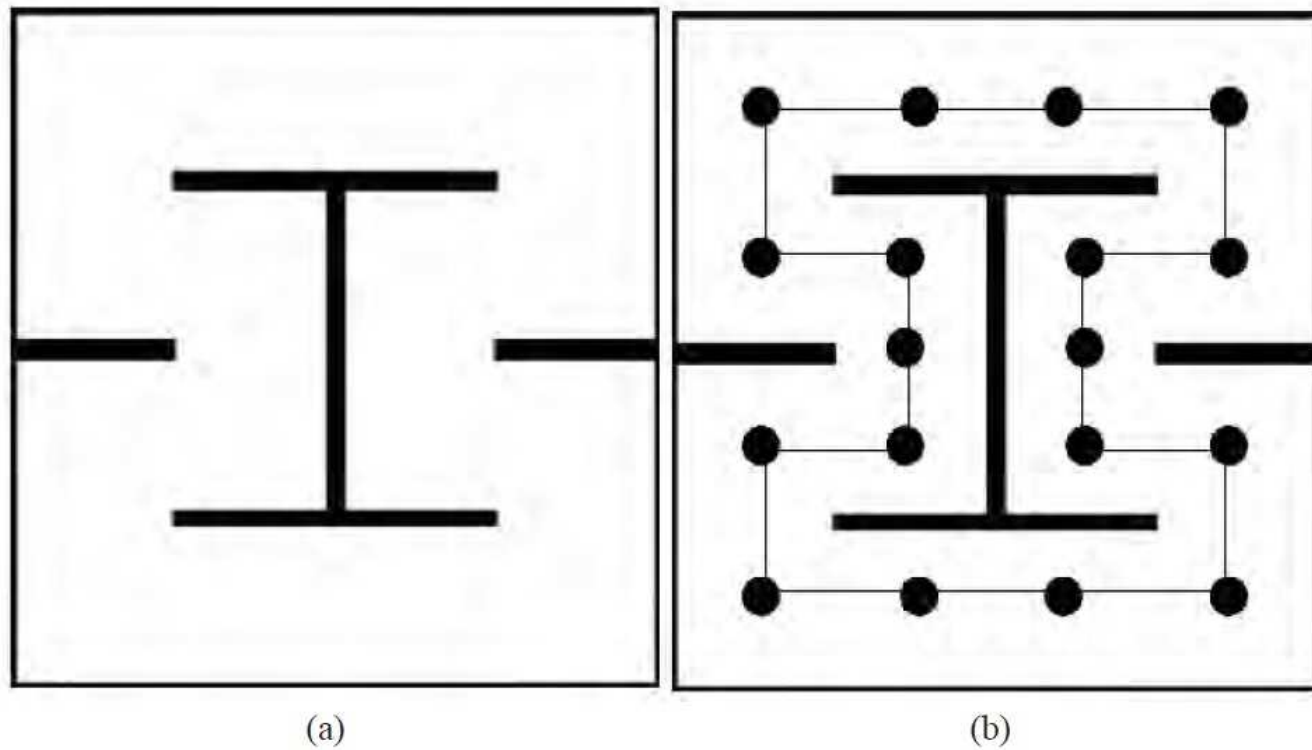
- Mapeamento = tarefa de construir representações de um ambiente, a partir dos dados obtidos dele.
- O robô deve ser capaz de perceber o ambiente a sua volta  
=> uso de sensores.
- Desafios:
  - Imprecisão e erros dos sensores;
  - Dimensão do ambiente;
  - Associação de dados;
  - Ambientes dinâmicos.



# Mapeamento

---

- Mapas topológicos ou relacionais



# Mapeamento

---

- Mapas métricos
  - Produz uma definição geométrica do ambiente em que o robô está inserido.
  - Mais fáceis de construir, representar e manter.
  - Requer uma determinação precisa da localização do robô.
  - Método utilizado neste trabalho: grades de ocupação

# Grades de Ocupação

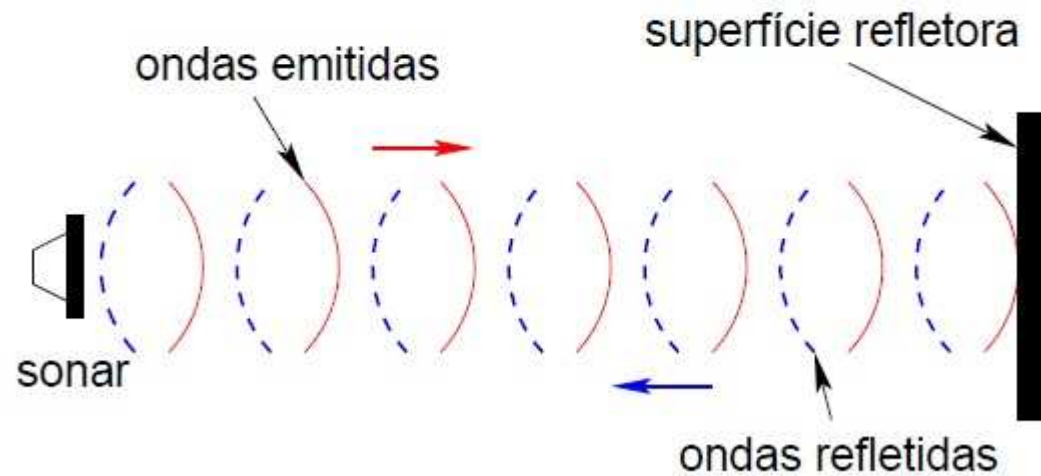
---

- Método proposto por Elfes (1987).
- Representação do ambiente em uma grade, ou matriz, de 2 ou 3 dimensões.
- Cada célula da grade armazena a probabilidade daquele espaço estar ocupado.
- A probabilidade de ocupação das células é atualizada considerando:
  - Teorema de Bayes;
  - Dados sensoriais, e seus modelos probabilísticos;
  - Localização do robô.

# Sensor de Ultrassom

---

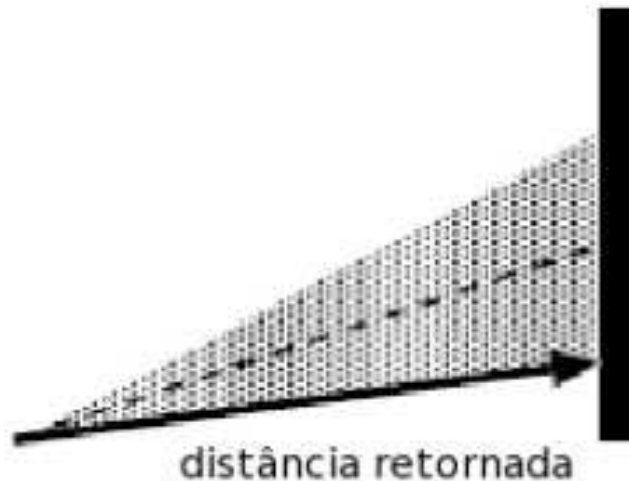
- Mede distâncias utilizando o tempo de propagação de ondas sonoras.



# Sensor de Ultrassom

---

- Problemas enfrentados:
  - Imprecisões no circuito temporizador
  - Distância reduzida



# Sensor de Ultrassom

---

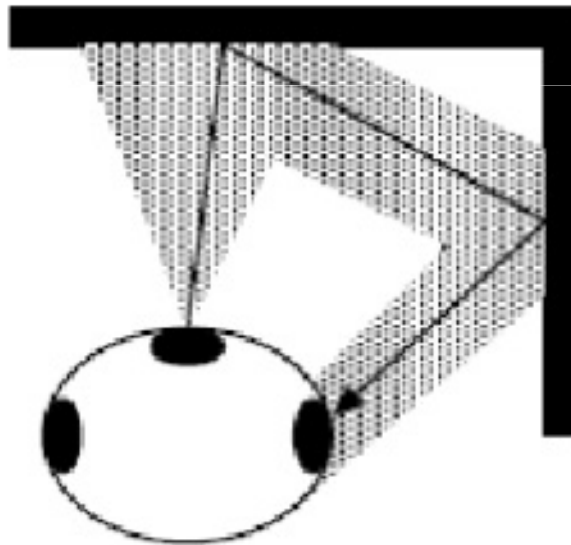
- Problemas enfrentados:
  - Reflexão especular



# Sensor de Ultrassom

---

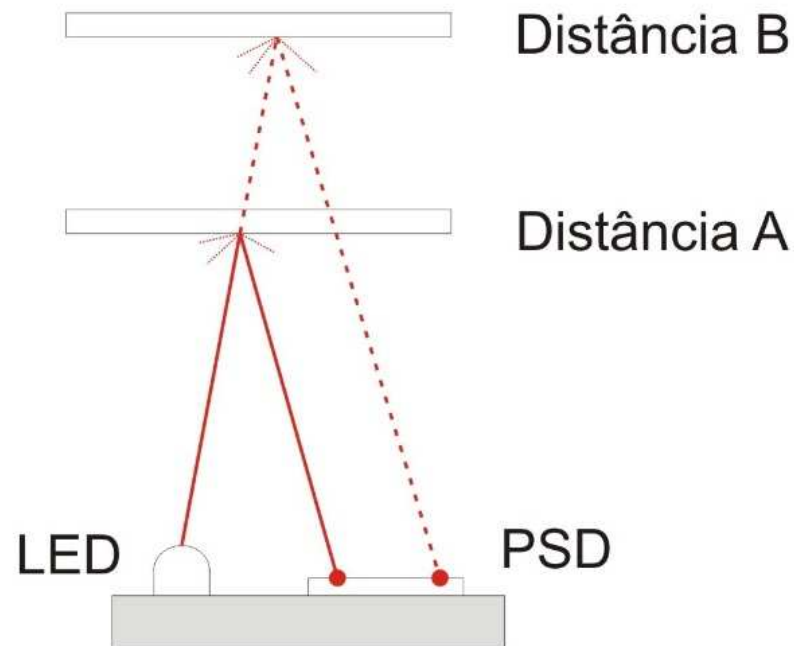
- Problemas enfrentados:
  - Leituras cruzadas



# Sensor Infravermelho

---

- Distância calculada por triangulação, considerando a posição em que o feixe de luz infravermelho atinge o dispositivo receptor PSD (*Position Sensing Device*).

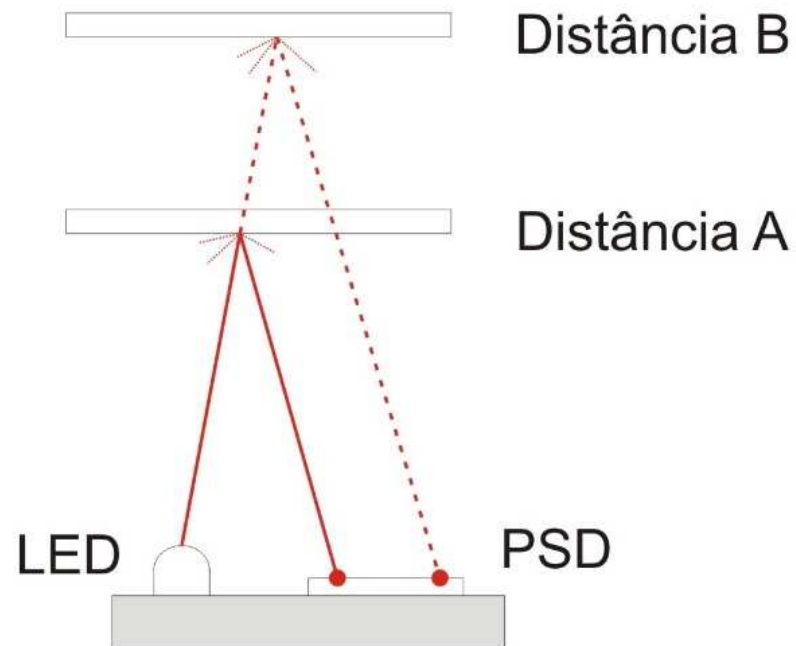




# Sensor Infravermelho

---

- Sujeito a erros semelhantes aos do sonar.
- Baixo custo e tamanho reduzido.



# Modelo probabilístico dos sensores

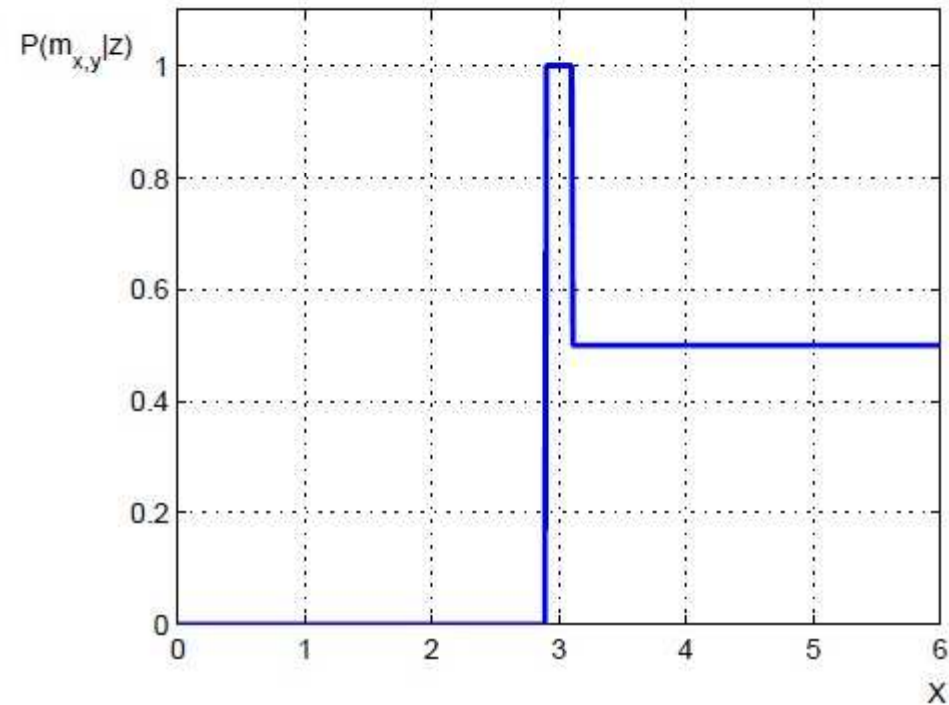
---

- Ambos sensores sujeitos a erros e imprecisões.
- Dificuldade em encontrar um modelo probabilístico que compreenda todas essas incertezas.
- O modelo probabilístico é necessário para a atualização das células da grade de ocupação.

# Modelo probabilístico dos sensores

---

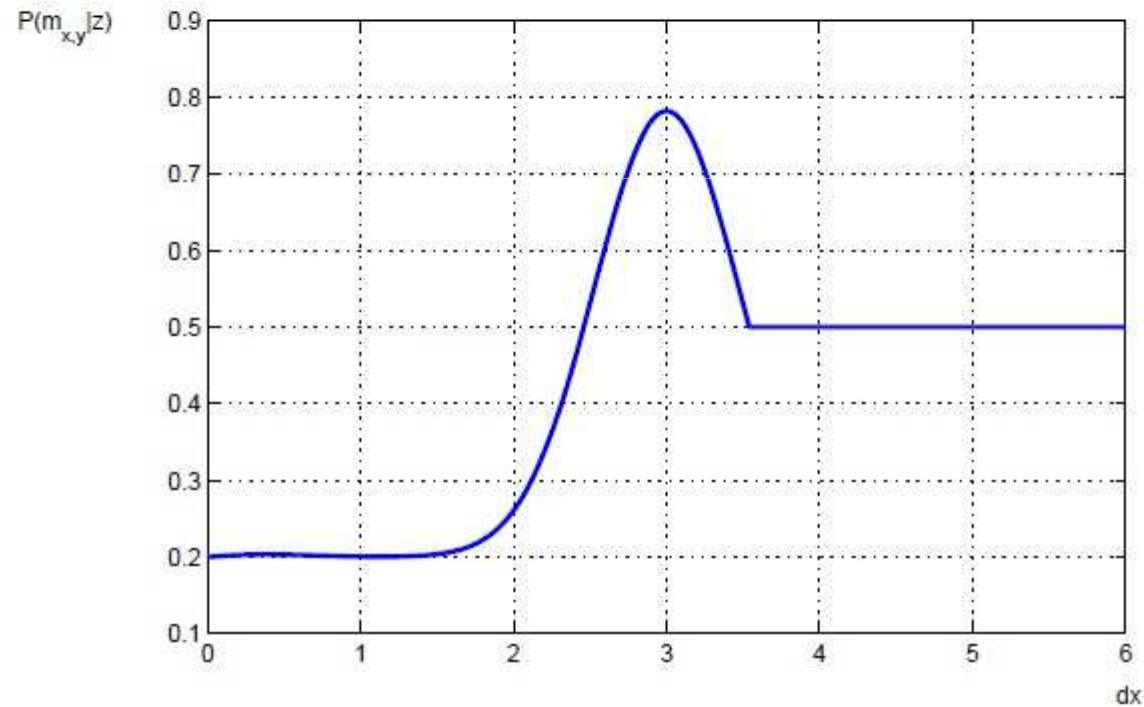
- Modelo de sensor ideal



# Modelo probabilístico dos sensores

---

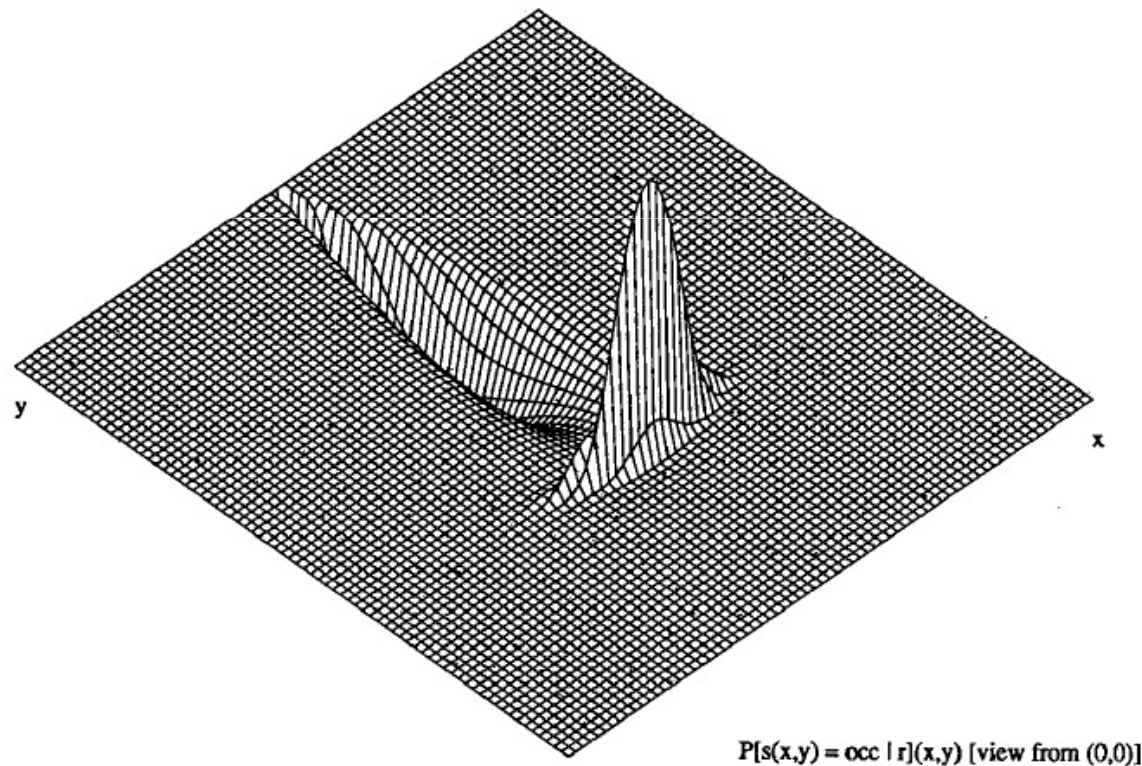
- Modelo utilizando distribuição Gaussiana



# Modelo probabilístico dos sensores

---

- Modelo utilizando distribuição Gaussiana



# SLAM

---

- Robô colocado em um ambiente desconhecido com localização desconhecida.
  - Para realizar o mapeamento é importante ter a localização.
  - Uma das formas de se localizar no ambiente é ter um mapa consistente dele, o qual pode ser consultado e comparado com os dados sensoriais obtidos.
- Um paradoxo se apresenta, o qual a solução é realizar o mapeamento e a localização simultaneamente, o que constitui o SLAM.
- Considerado um problema difícil que enfrenta diversos desafios.

# SLAM

---

- Problemas e desafios:
  - Ambientes dinâmicos;
  - Associação de dados;
  - Estratégia de exploração;
  - Dimensionalidade;
  - Erros gerados pelos sensores.

# SLAM

---

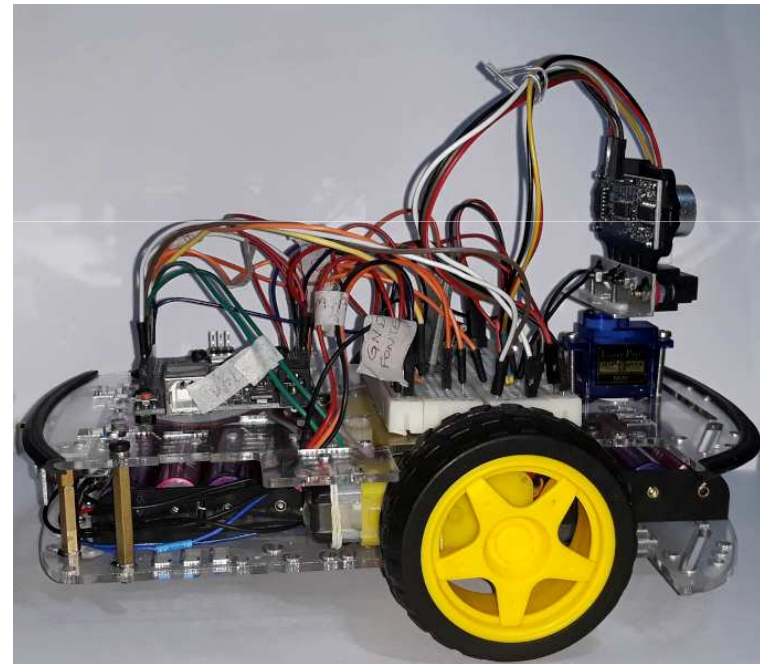
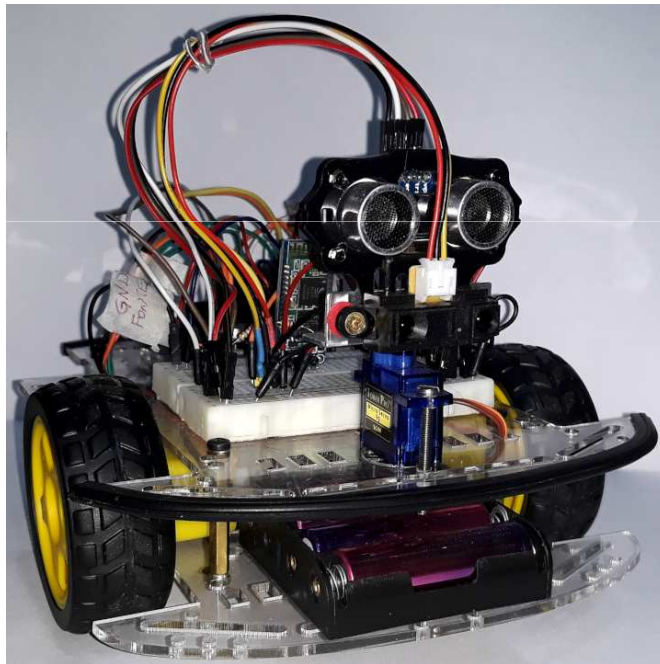
- As soluções vão envolver os métodos já discutidos de localização e mapeamento.
- Exemplos de soluções:
  - EKF-SLAM
  - FastSLAM
  - GridSLAM
  - DP-SLAM



# Hardware

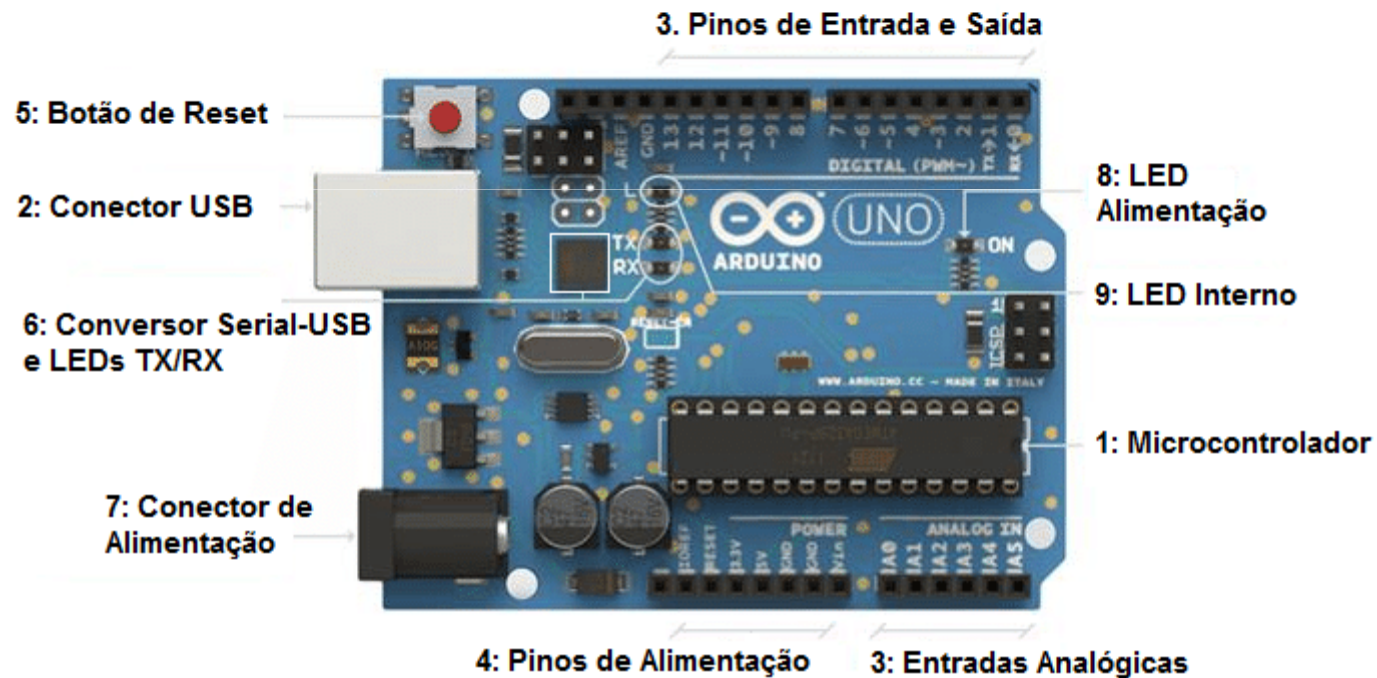
---

- Robô Frank.



# Hardware

## ■ Arduino.



# Hardware

---

- Sensores.



# Hardware

---

- Sensores.



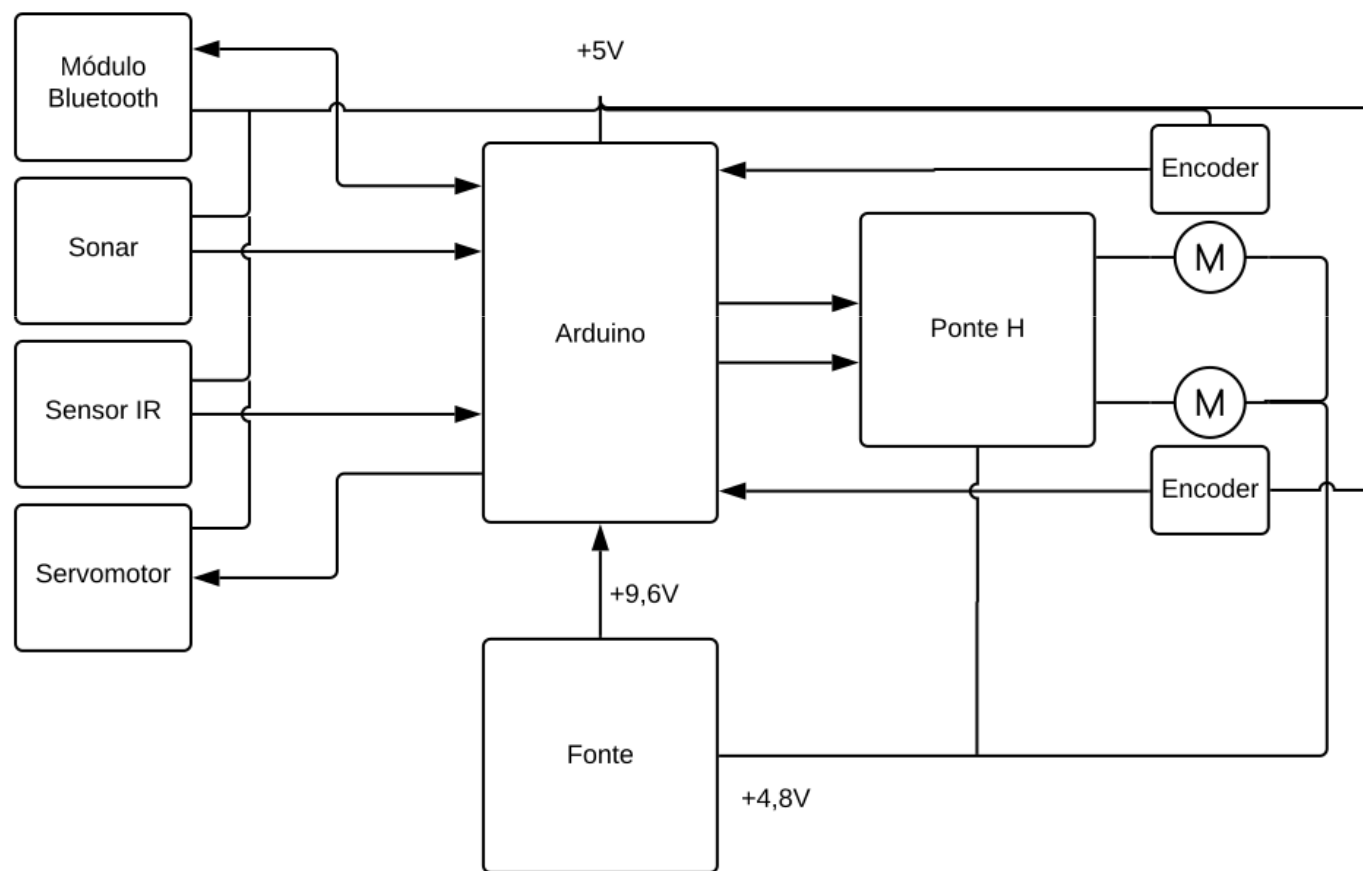
# Hardware

---

- Outros componentes.



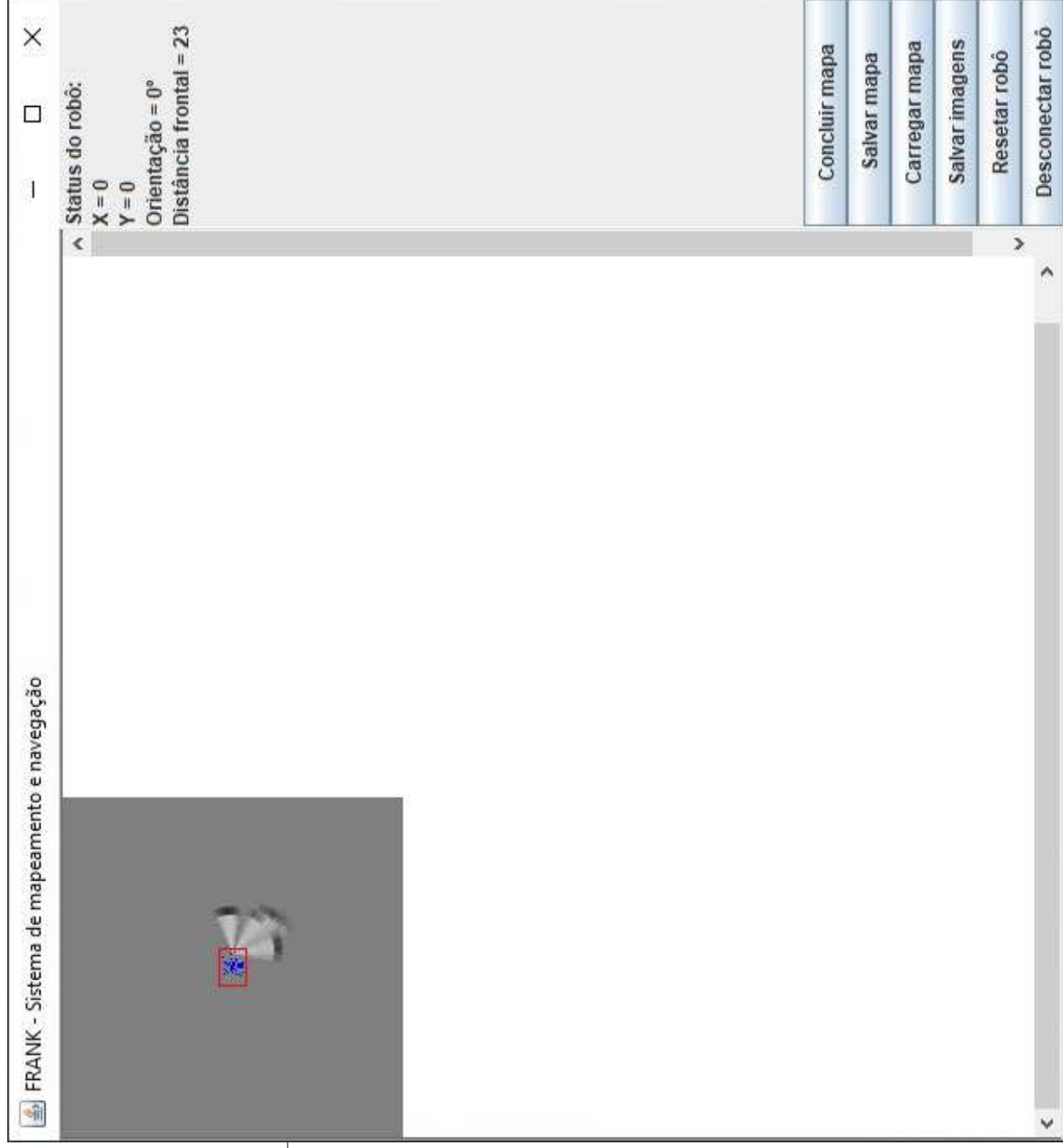
# Hardware



# Software

---

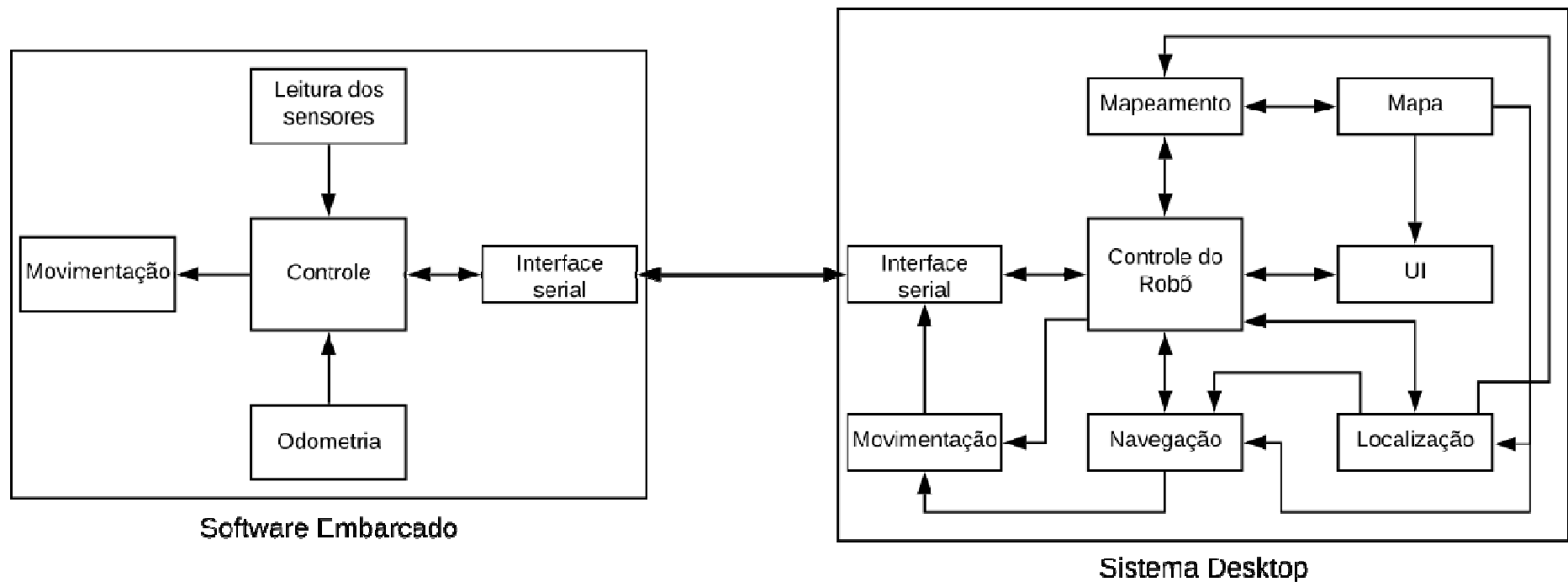
- Dividido em dois sistemas:
  - Sistema embarcado no Arduino (C++);
  - Sistema *Desktop* (Java).
- O sistema desenvolvido permite:
  - Mapear o ambiente do robô;
  - Navegar entre dois pontos criando uma rota a partir do mapa;
  - Controlar a movimentação do robô manualmente;
  - Salvar e carregar mapas produzidos.





# Software

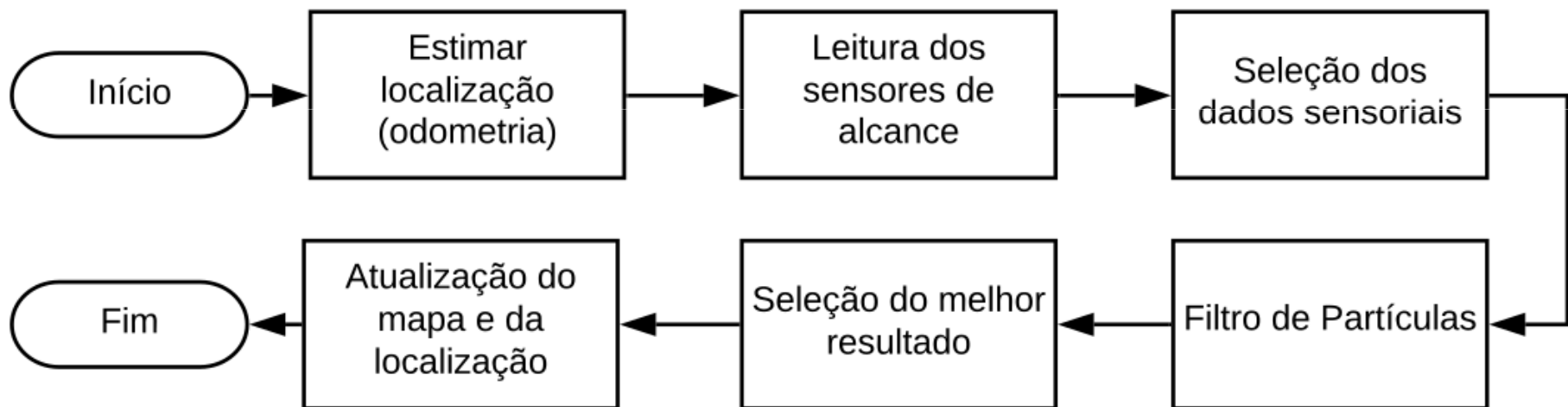
## ■ Arquitetura geral:



# Software

---

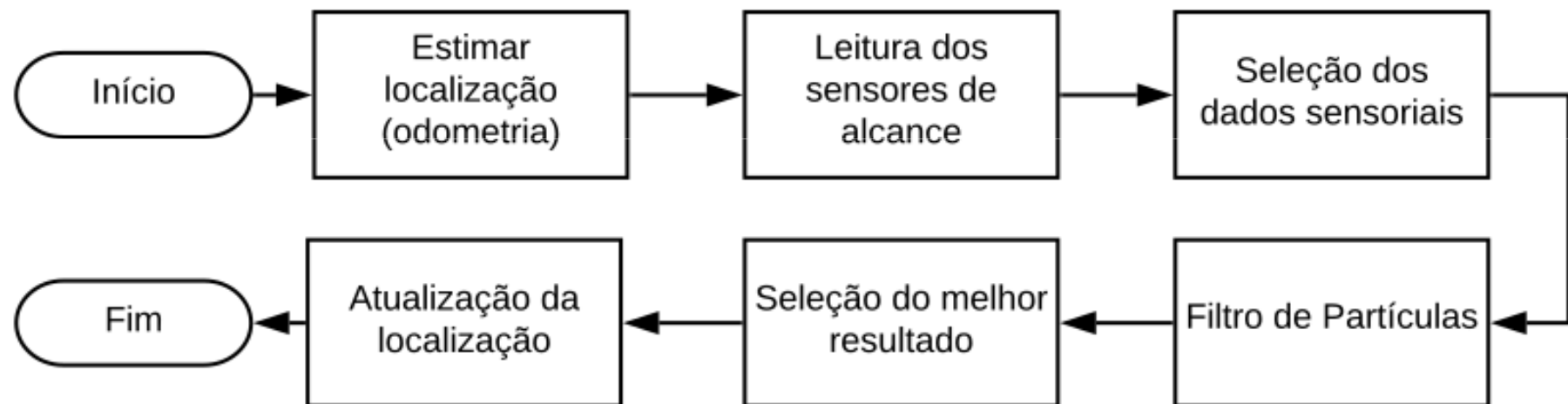
## ■ Mapeamento:



# Software

---

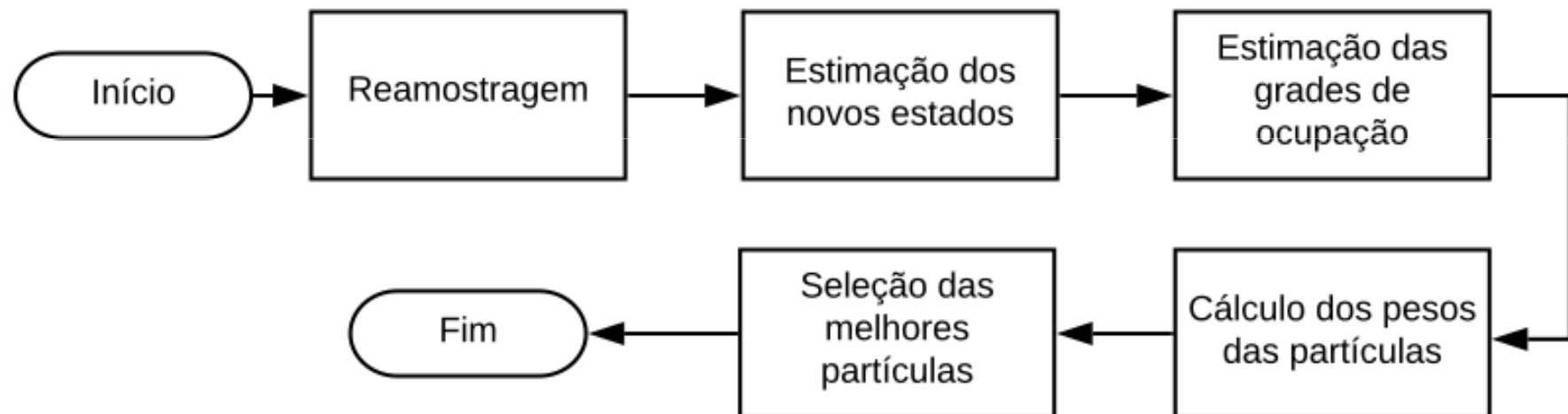
## ■ Localização:



# Software

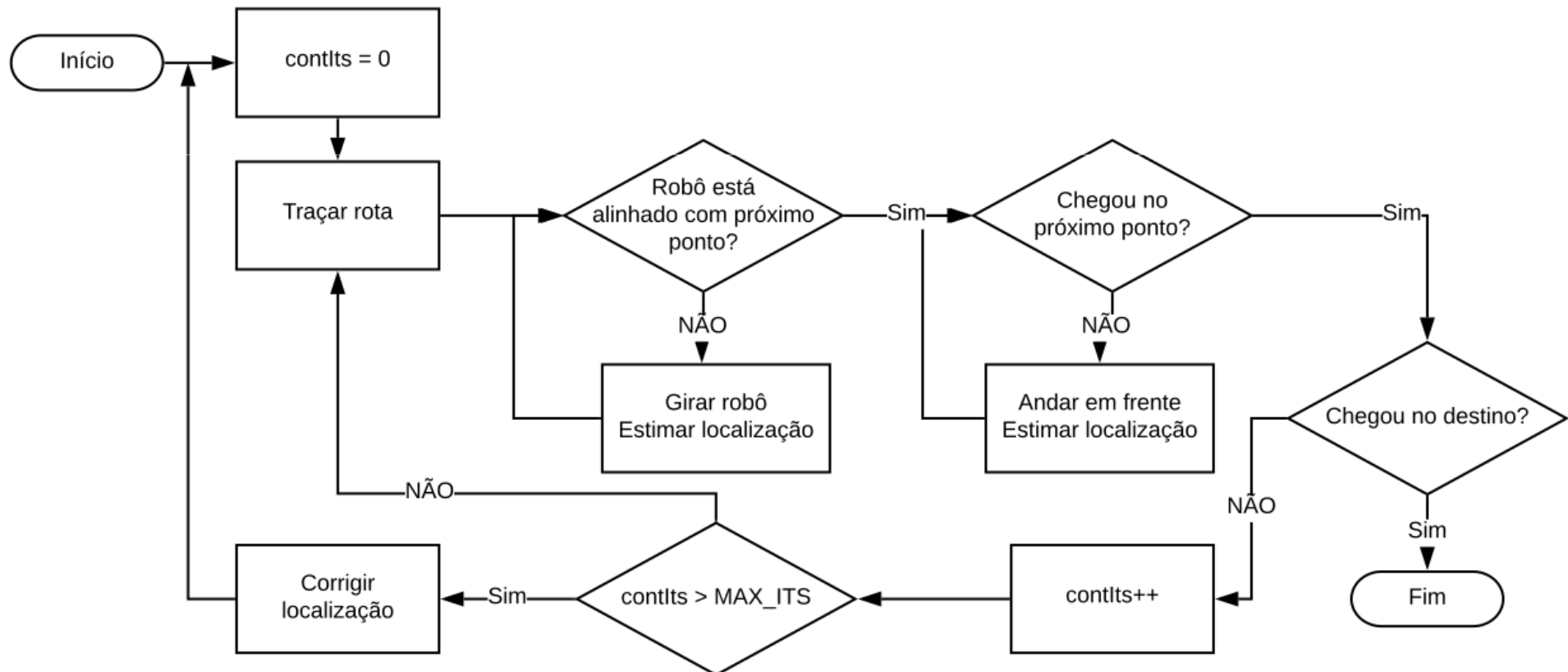
---

## ■ Filtro de Partículas:



# Software

## ■ Navegação:



# Resultados

---

- Erros de odometria:
  - Foram realizados experimentos para aferir a precisão da odometria.
  - Os maiores erros encontrados foram na estimativa da orientação do robô.
  - Isso praticamente inviabiliza o uso do filtro de Kalman para estimar a localização do robô (BONTEMPO, 2011).

# Resultados

---

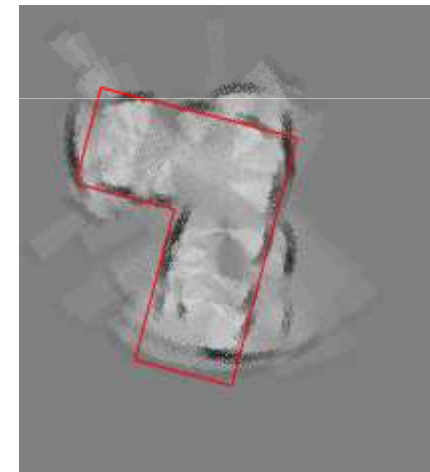
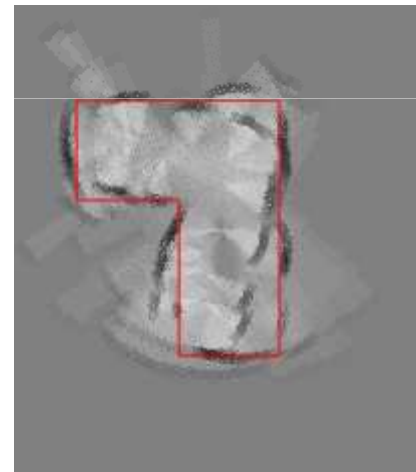
- Ambiente para testes de mapeamento e navegação



# Resultados

---

- Mapeamento sem filtro de partículas (apenas odometria)

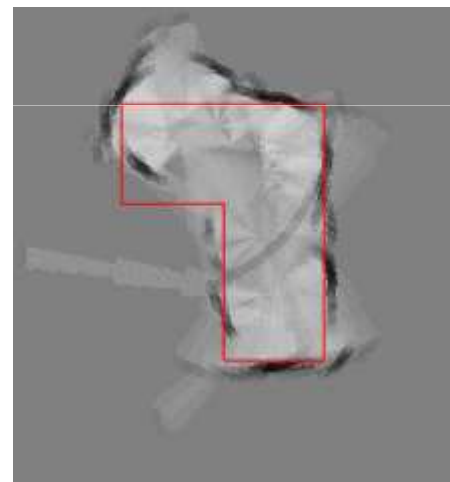




# Resultados

---

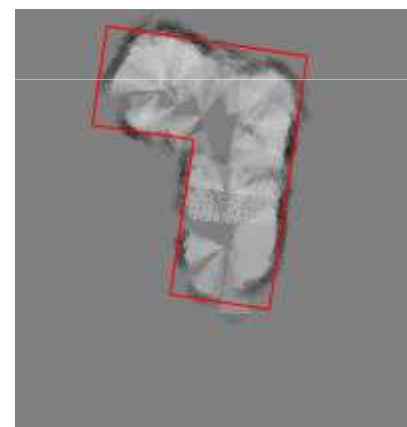
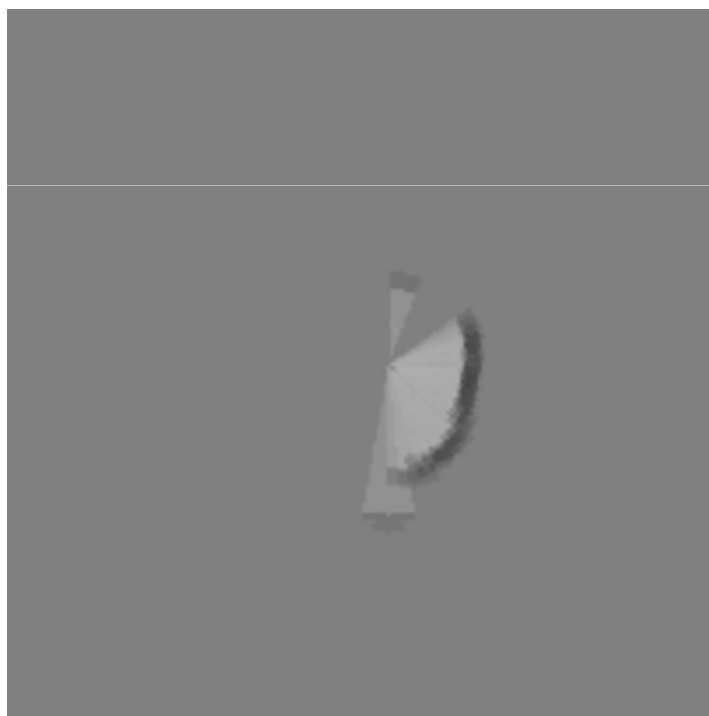
- Mapeamento utilizando apenas dados do sonar.



# Resultados

---

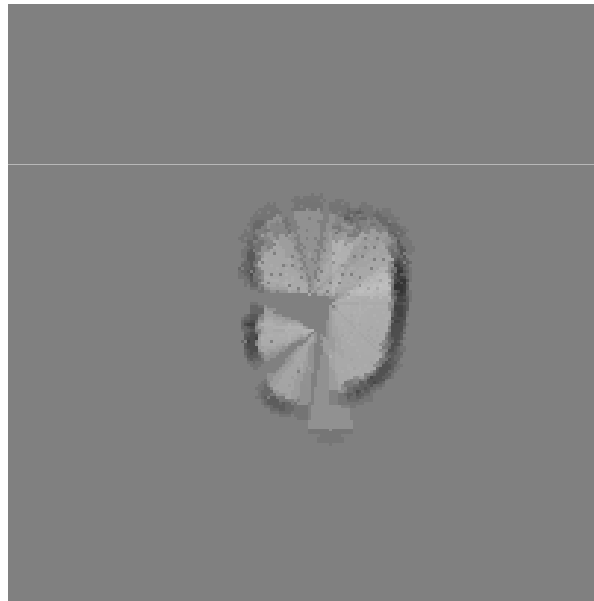
- Mapeamento utilizando apenas dados do infravermelho.



# Resultados

---

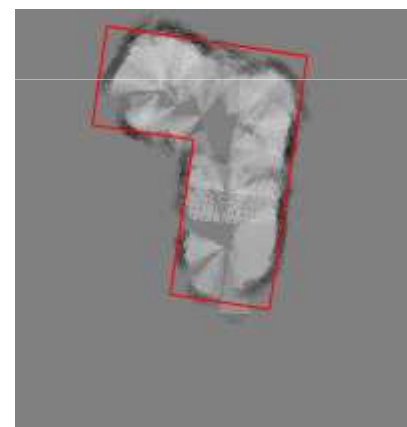
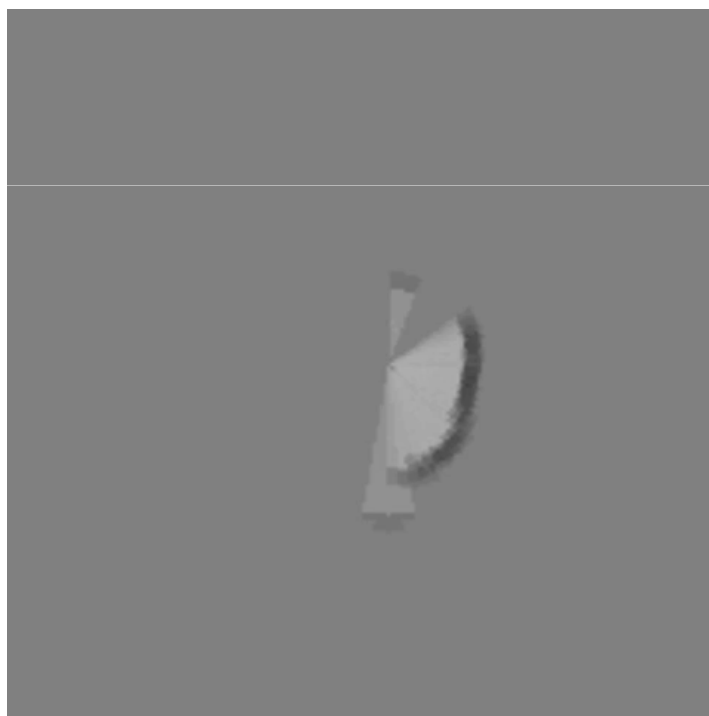
- Mapeamento utilizando apenas dados do infravermelho.



# Resultados

---

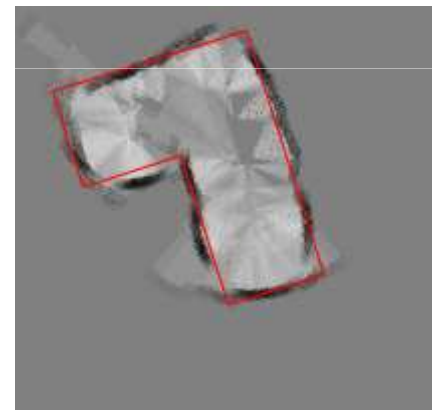
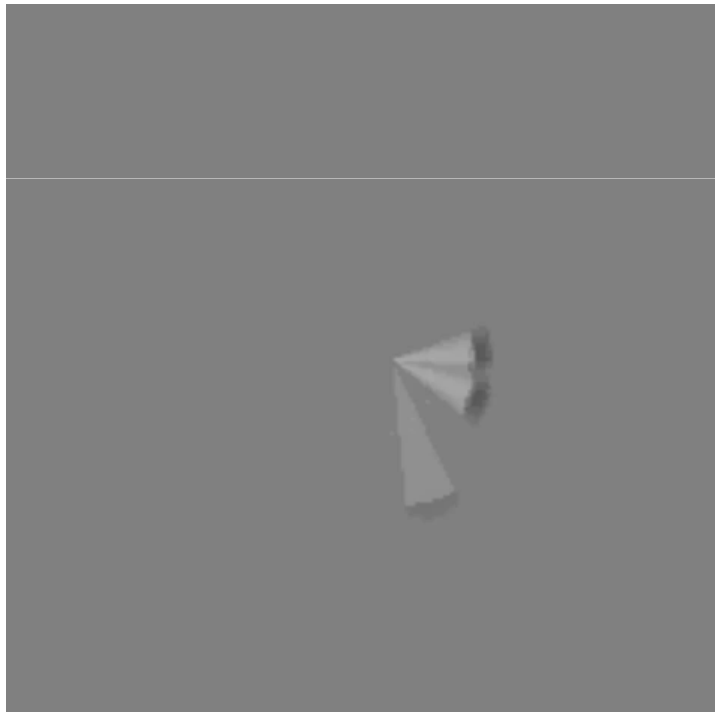
- Mapeamento utilizando apenas dados do infravermelho.



# Resultados

---

- Mapeamento utilizando ambos os sensores.



# Resultados

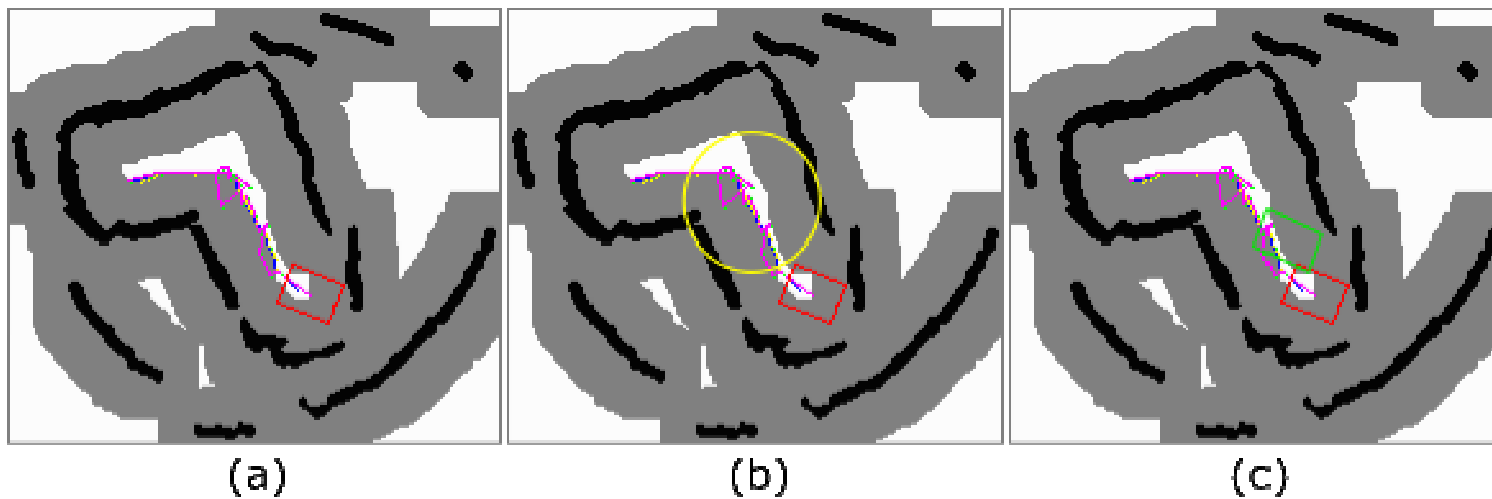
---

- Os resultados utilizando ambos os sensores são relativamente melhores que os dos outros experimentos.
- Os resultados obtidos são satisfatórios, para fins do estudo do problema, ainda mais considerando que são utilizados apenas sensores de baixo custo, com uma taxa de erro alta.

# Resultados

---

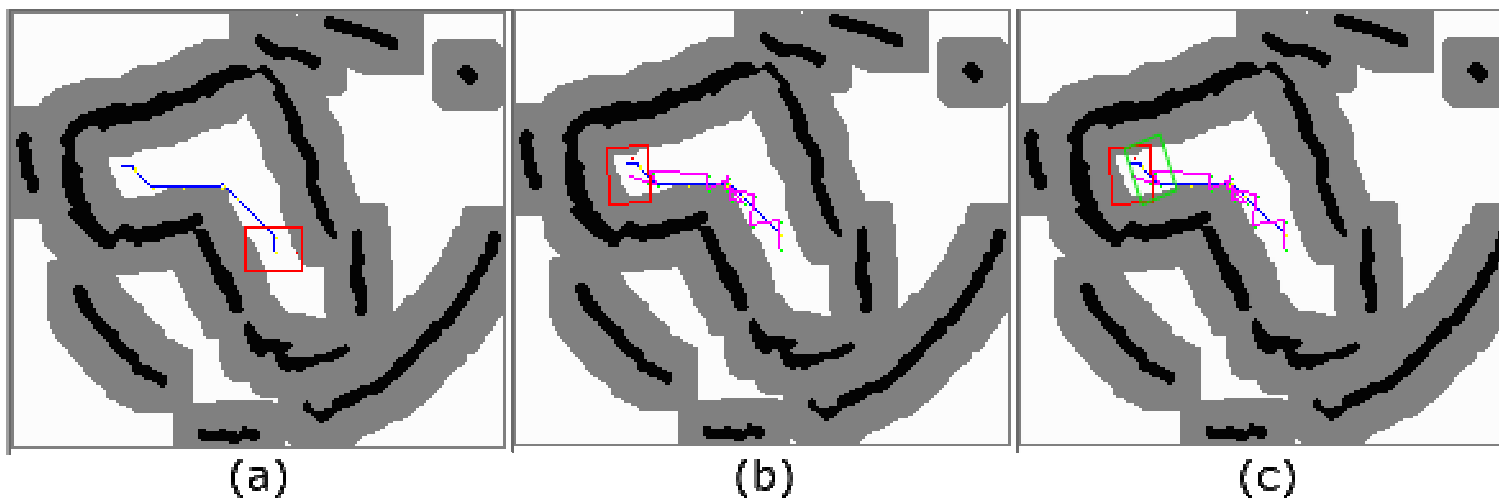
- Resultados da navegação



# Resultados

---

- Resultados da navegação

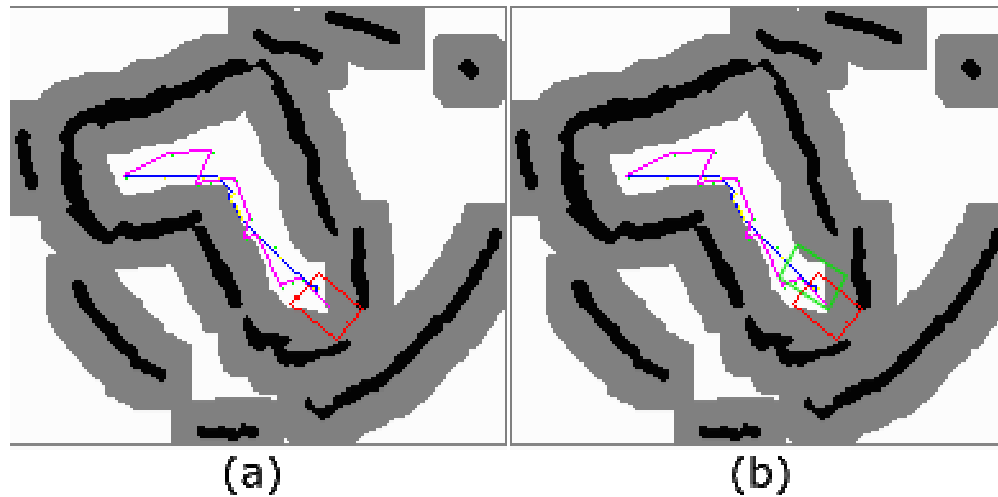




# Resultados

---

- Resultados da navegação



# Resultados

---

- Os resultados da navegação também foram satisfatórios, para fins de estudos do problema.
- O filtro de partículas para a correção da localização se mostrou essencial para o sucesso da navegação.
- Problemas apresentados:
  - Possíveis colisões com as laterais e a traseira do robô, devido a falta de sensores nessas posições;
  - Tempo de navegação.

# Conclusões

---

- Resultados finais são satisfatórios, do ponto de vista de estudo do problema, em especial pela utilização de sensores de baixo custo.
- Limitação e imprecisão dos sensores e da odometria.
- Os resultados apontam a possibilidade da construção e utilização de robôs de custos não tão elevados para a realização de tarefas complexas.
- Diversas possibilidades de melhorias

# Conclusões

---

- Trabalhos futuros:
  - Utilização de sensores melhores e/ou em maior quantidade;
  - Aplicação de técnicas mais robustas de fusão sensorial;
  - Adição de uma bússola ao robô, melhorando a estimativa da orientação e permitindo a utilização de outras soluções, como o filtro de Kalman;
  - Melhoria da comunicação entre o robô e o computador;
  - Melhoria da método de localização;
  - Implementação de algoritmo de exploração;
  - Teste de métodos baseados em *landmarks*;
  - Processamento paralelo.

# Referências

---

BIGHETI, J. A. Navegação de robôs em ambientes internos usando SLAM. Dissertação (Mestrado) — Universidade Estadual Paulista, 2011. Disponível em: <<http://hdl.handle.net/11449/87178>>. Acesso em: 06 mar. 2019.

BONTEMPO, A. P. Uma Abordagem Híbrida para Localização e Mapeamento Simultâneos para Robôs Móveis com Sonares Através de Filtro de Kalman Estendido. Dissertação (Mestrado) — PUC-Rio, 2012

ELFES, A. Sonar-based real-world mapping and navigation. IEEE Journal on Robotics and Automation, IEEE, v. 3, n. 3, p. 249–265, 1987.

MARCHI, J. Navegação de robôs móveis autônomos: estudo implementação de abordagens. Florianópolis, SC, 2001. Disponível em: <<https://repositorio.ufsc.br/xmlui/handle/123456789/81441>>. Acesso em: 06 mar. 2019.

# Referências

---

MATARIĆ, M. J. The robotics primer. Cambridge: Mit Press, 2007.

SOUZA, A. A. d. S. Mapeamento com sonar usando grade de ocupação baseado em modelagem probabilística. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, 2008.