

**Universidade Tecnológica Federal do Paraná**  
Engenharia de Computação  
*Campus Pato Branco*

**Trabalho de Conclusão de Curso - 2019/1**  
Navegação de robôs móveis utilizando Lógica *Fuzzy*

Profa. Orientadora: Dra. Kathya Silvia Collazos Linares  
Acadêmico: Marcelo Gervazoni Carbonera

2 de Julho de 2019

# Estrutura

- 1. Introdução
- 2. Objetivos
- 3. Fundamentos Teóricos
  - 3.1 Modelo do robô móvel
  - 3.2 Sistemas a Eventos Discretos
  - 3.3 Sistemas Híbridos
  - 3.4 Lógica *Fuzzy* em Controle
- 4. Materiais e Método
- 5. Desenvolvimento
- 6. Considerações

# Introdução

## Aspectos Qualitativos

### Arquiteturas:

- Deliberativa.
- Reativa.
- Híbrida.
- Comportamental.

# Introdução

## Aspectos Qualitativos

### Arquiteturas:

- Deliberativa.
- Reativa.
- Híbrida.
- Comportamental.

### Deliberativa

“Pensar antes de agir”.

Foco no longo prazo.

# Introdução

## Aspectos Qualitativos

### Arquiteturas:

- Deliberativa.
- Reativa.
- Híbrida.
- Comportamental.

### Reativa

Reagir rapidamente a estímulos, sem planejar.

Foco no curto prazo.

# Introdução

## Aspectos Qualitativos

### Arquiteturas:

- Deliberativa.
- Reativa.
- Híbrida.
- Comportamental.

## Híbrida

Camadas deliberativa e reativa em conjunto.

Desafio: extrair o melhor de cada abordagem.

# Introdução

## Aspectos Qualitativos

### Arquiteturas:

- Deliberativa.
- Reativa.
- Híbrida.
- Comportamental.

## Comportamental

Semelhante à arquitetura reativa.

Difere pelo nível de abstração.

# Objetivos

## Objetivo Geral

- Construir robô autônomo, sem armazenamento de mapa, capaz de desviar de obstáculos.
- Comparar qualitativamente: controlador Híbrido e *Fuzzy*.

## Objetivos Específicos

- Modelagem.
- Arquitetura de controle.
- Implementação física.
- Implementação do sistema embarcado.
- Teste e validação.



# Objetivos

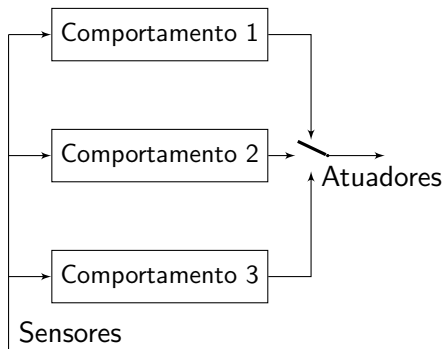
## Objetivo Geral

- Construir robô autônomo, sem armazenamento de mapa, capaz de desviar de obstáculos.
- Comparar qualitativamente: controlador Híbrido e *Fuzzy*.

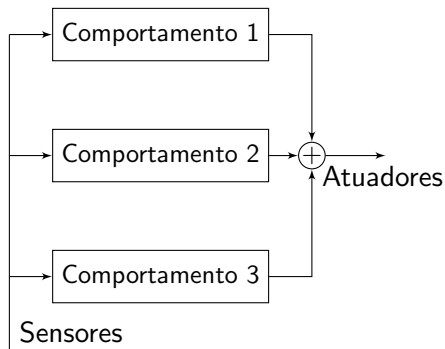
## Objetivos Específicos

- Modelagem.
- Arquitetura de controle.
- Implementação física.
- Implementação do sistema embarcado.
- Teste e validação.

# Arquitetura comportamental

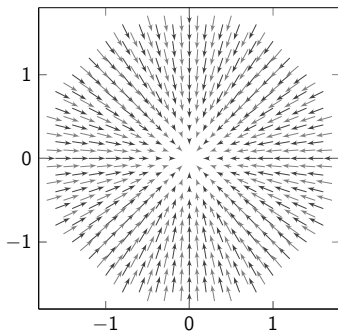


(a) Arbitragem

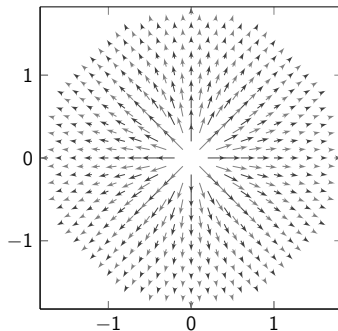


(b) Fusão

# Comportamentos como campos potenciais



(a) Comportamento “Ir Para Objetivo”

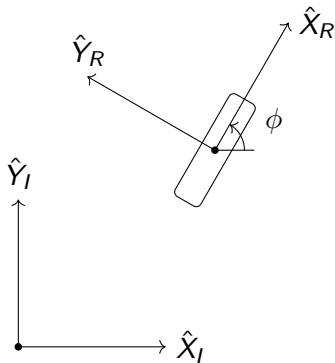


(b) Comportamento “Evitar Obstáculo”

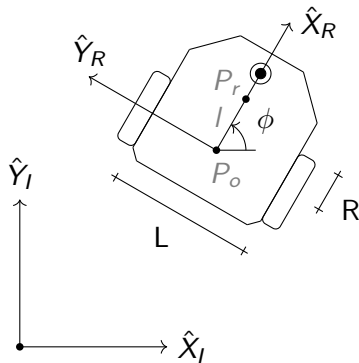
## Mínimos locais

- Existência de mínimos locais depende da disposição e formato de obstáculos.
- Um comportamento “Seguidor de parede” pode ser criado a fim de contornar obstáculos.

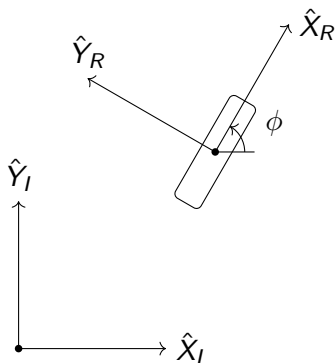
# Modelo Cinemático



(a) Uniciclo



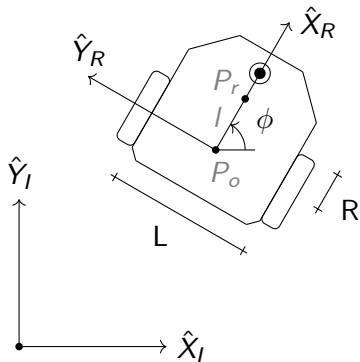
(b) Diferencial



$$\begin{cases} \dot{x} &= v \cos \phi \\ \dot{y} &= v \sin \phi \\ \dot{\phi} &= \omega \end{cases}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos \phi & 0 \\ \sin \phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

$$\begin{cases} \dot{x} = \frac{R}{2}(\omega_l + \omega_r) \cos \phi \\ \dot{y} = \frac{R}{2}(\omega_l + \omega_r) \sin \phi \\ \dot{\phi} = \frac{R}{L}(\omega_r - \omega_l) \end{cases}$$



# Estabelecendo vínculo entre os modelos

$$\begin{cases} \dot{x} = \frac{R}{2}(\omega_l + \omega_r) \cos \phi \\ \dot{y} = \frac{R}{2}(\omega_l + \omega_r) \sin \phi \\ \dot{\phi} = \frac{R}{L}(\omega_r - \omega_l) \end{cases}$$

$$\begin{cases} \dot{x} = v \cos \phi \\ \dot{y} = v \sin \phi \\ \dot{\phi} = \omega \end{cases}$$

## Equivalência entre modelos

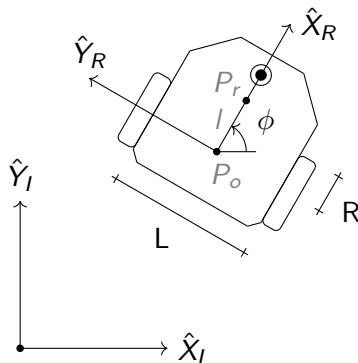
$$\omega_l = \frac{2v - L\omega}{2R}$$

$$\omega_r = \frac{2v + L\omega}{2R}$$



# O modelo de integrador único

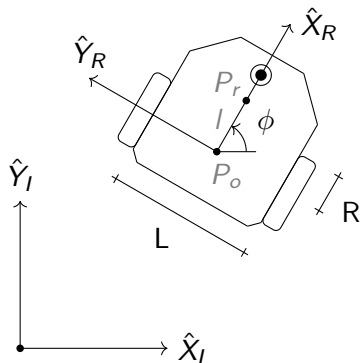
$$\begin{cases} x' = x_o + l \cos \phi \\ y' = y_o + l \sin \phi \end{cases}$$



# O modelo de integrador único

$$\begin{cases} x' = x_o + l \cos \phi \\ y' = y_o + l \sin \phi \end{cases}$$

$$\begin{cases} \dot{x}' = \dot{x}_o - \dot{\phi} l \sin \phi = v \cos \phi - l \omega \sin \phi \\ \dot{y}' = \dot{y}_o + \dot{\phi} l \cos \phi = v \sin \phi + l \omega \cos \phi \end{cases}$$

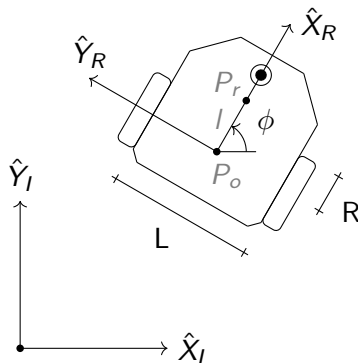


# O modelo de integrador único

$$\begin{cases} x' = x_o + l \cos \phi \\ y' = y_o + l \sin \phi \end{cases}$$

$$\begin{cases} \dot{x}' = \dot{x}_o - \dot{\phi} l \sin \phi = v \cos \phi - l \omega \sin \phi \\ \dot{y}' = \dot{y}_o + \dot{\phi} l \cos \phi = v \sin \phi + l \omega \cos \phi \end{cases}$$

$$\begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix} = \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & l \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$



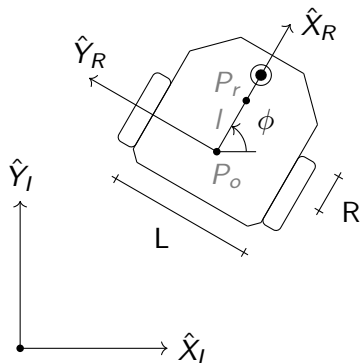
# O modelo de integrador único

$$\begin{cases} x' = x_o + l \cos \phi \\ y' = y_o + l \sin \phi \end{cases}$$

$$\begin{cases} \dot{x}' = \dot{x}_o - \dot{\phi} l \sin \phi = v \cos \phi - l \omega \sin \phi \\ \dot{y}' = \dot{y}_o + \dot{\phi} l \cos \phi = v \sin \phi + l \omega \cos \phi \end{cases}$$

$$\begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix} = \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & l \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{l} \end{bmatrix} \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix}$$



# O modelo de integrador único

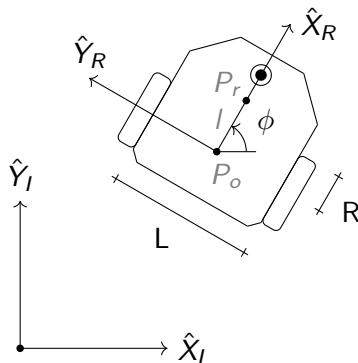
$$\begin{cases} x' = x_o + l \cos \phi \\ y' = y_o + l \sin \phi \end{cases}$$

$$\begin{cases} \dot{x}' = \dot{x}_o - \dot{\phi} l \sin \phi = v \cos \phi - l \omega \sin \phi \\ \dot{y}' = \dot{y}_o + \dot{\phi} l \cos \phi = v \sin \phi + l \omega \cos \phi \end{cases}$$

$$\begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix} = \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & l \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{l} \end{bmatrix} \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix}$$

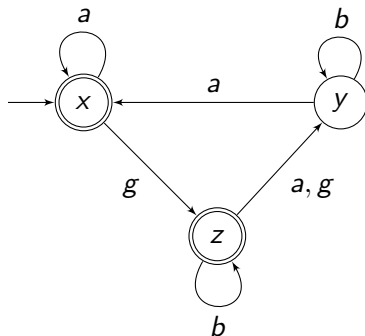
$$\begin{bmatrix} \omega_l \\ \omega_r \end{bmatrix} = \begin{bmatrix} \frac{1}{R} & \frac{-L}{2R} \\ \frac{1}{R} & \frac{L}{2R} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{l} \end{bmatrix} \begin{bmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \dot{x}' \\ \dot{y}' \end{bmatrix}$$



# Sistemas a Eventos Discretos

## Definição

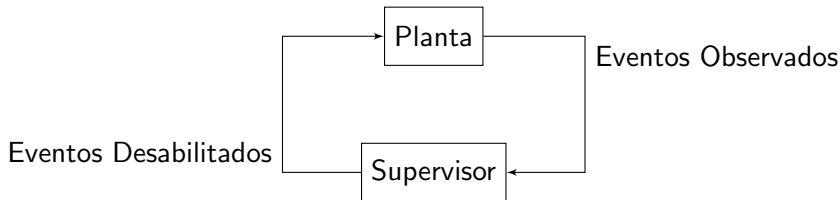
- Sistema Dinâmico.
- Estados discretos.
- Eventos assíncronos provocam transições instantâneas de estado.
- Arbitragem!



# Controle Supervisório

## Definição

- Captura eventos observáveis.
- Permite ou suprime eventos passíveis de ocorrência.



# Automatos Temporizados com guarda

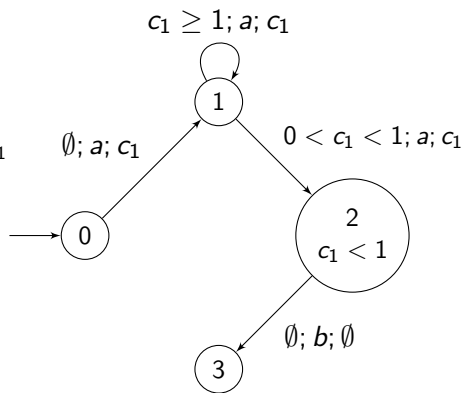
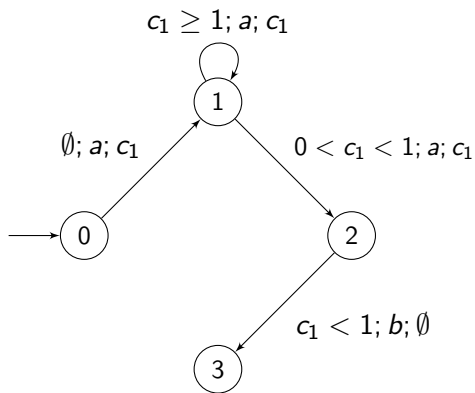
## *clocks*

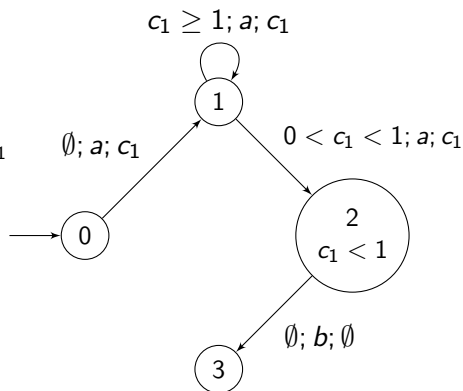
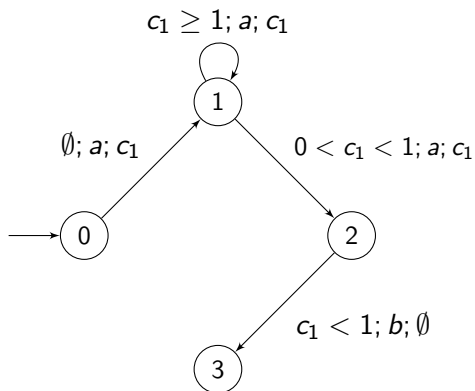
- “Contador”: associado a tempo.
- Definido por uma função linear ( $\dot{x} = 1$ ).

## Função de transição

- Transições: (Estado, Evento)  $\rightarrow$  (Estado)
- Transições Temporizadas: (Guardas, Evento, *Reset*)  $\rightarrow$  (Estado)
  - $\rightarrow$  Guardas: condições nas variáveis contínuas.
  - $\rightarrow$  *Reset*: conjunto de variáveis *clock* às quais deve-se atribuir valor zero.







## Invariante

Evita *deadlock* por tempo ao forçar transição!

# Sistemas Híbridos

## Definição

- Estados formados por tuplas  $(q, \mathbf{x})$ .
  - $q$ : Estados discretos .
  - $\mathbf{x}$ : Estados contínuos .
- Autômato híbrido estende o autômato temporizado com guardas.
- Estados são chamados “**modos de operação**”, pois alteram a dinâmica do sistema.

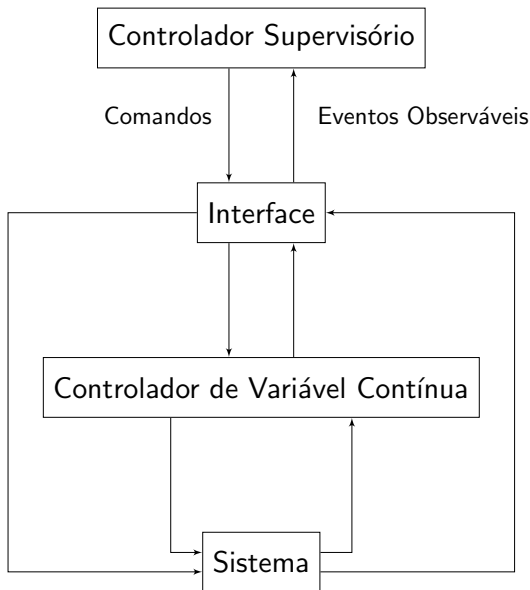
# Sistemas Híbridos

## Definição

- Estados formados por tuplas  $(q, \mathbf{x})$ .
  - $q$ : Estados discretos .
  - $\mathbf{x}$ : Estados contínuos .
- Autômato híbrido estende o autômato temporizado com guardas.
- Estados são chamados “**modos de operação**”, pois alteram a dinâmica do sistema.

## Arbitragem

Chaveamento de dinâmicas (controladores).

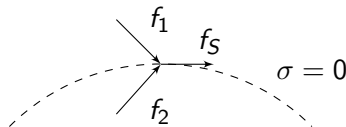


## Campos vetoriais descontínuos

- Regiões de costura.
- Deslizamento.
- Comportamento de Zenão.
- Regularização.

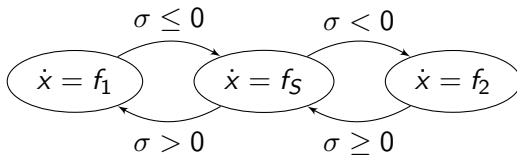
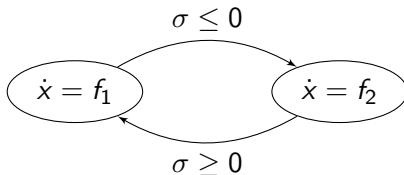
## Campos vetoriais descontínuos

- Regiões de costura.
- Deslizamento.
- Comportamento de Zenão.
- Regularização.



## Campos vetoriais descontínuos

- Regiões de costura.
- Deslizamento.
- Comportamento de Zenão.
- Regularização.





# Lógica *Fuzzy* em controle

## Conjuntos *fuzzy*

- Variável linguística.
- Valor linguístico.
- Grau de pertinência.

# Lógica Fuzzy em controle

## Conjuntos fuzzy

- Variável linguística.
- Valor linguístico.
- Grau de pertinência.

## Motivação para uso

- Mescla aspectos exatos e elementos qualitativos.

# Lógica *Fuzzy* em controle

## Conjuntos *fuzzy*

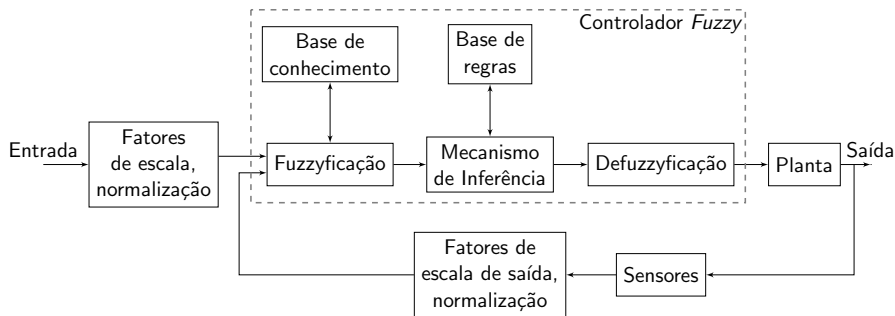
- Variável linguística.
- Valor linguístico.
- Grau de pertinência.

## Motivação para uso

- Mescla aspectos exatos e elementos qualitativos.

## Indicação e contraindicação

- Indicada em sistemas **complexos**, compreendidos parcialmente.
- Deseja-se solução rápida.
- Não apropriada para sistemas simples (LIT).



## Fuzzyficação

- Associa variáveis numéricas a conjuntos *fuzzy*.
- Saída: conjunto de graus de pertinência vinculados a valores linguísticos.

## Fuzzyficação

- Associa variáveis numéricas a conjuntos *fuzzy*.
- Saída: conjunto de graus de pertinência vinculados a valores linguísticos.

## Mecanismo de Inferência

- Base de regras ( $P \rightarrow Q$ ).
- Pertinência das premissas determinam pertinências das saídas.
- Conjunto de “recomendações”.

## Fuzzyficação

- Associa variáveis numéricas a conjuntos *fuzzy*.
- Saída: conjunto de graus de pertinência vinculados a valores linguísticos.

## Mecanismo de Inferência

- Base de regras ( $P \rightarrow Q$ ).
- Pertinência das premissas determinam pertinências das saídas.
- Conjunto de “recomendações”.

## Defuzzificação

- A partir de recomendações distintas, determinam-se as saídas.

# Materiais

## Tabela de Custos

Item	Qtd	Custo Unitário	Custo Total
Bateria LiPo Limskey 11,1v 3S 4200mAh	1	112,05	112,05
Módulo para cartão de memória SD WAVGAT, comunicação SPI	1	2,79	2,79
Ponte H dupla L298N	1	6,12	6,12
Sensor infravermelho GP2Y0A21YK0F, alcance de 10 a 80 cm	5	13,07	65,35
Módulo conversor Buck DC-DC, saída 5v	1	9,14	9,14
Sensor inercial IMU GY-87, com 10 graus de liberdade (MPU6050, HCM5883L, BMP180)	1	25,9	25,9
Módulo wireless NRF24L01 com antena de alcance de 1000 metros	1	7,69	7,69
Conjuntos de motores, caixa de redução 1:34 e sensores de efeito hall (341,2 PPR)	2	47,015	94,03
Esfera de rolagem	1	2,002	2,002
Tiva Connected LaunchPad TM4C1294	1	80,00	80,00
		Total:	405,07



# Metodologia

- Projetar controladores.
- Validar o funcionamento usando simulador Simiam.
- Montagem física.
- Sistema embarcado.
- Teste e validação.

# Desenvolvimento

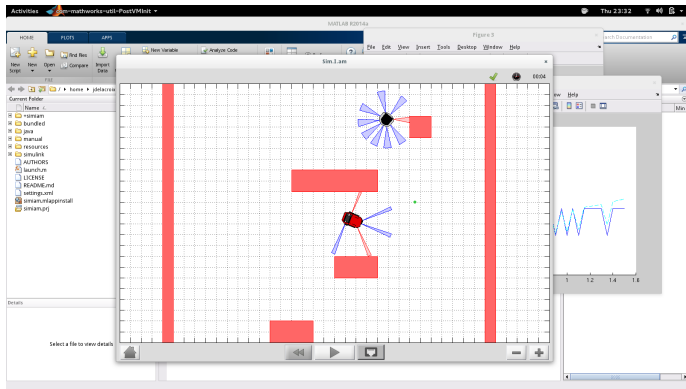
## Simiam

- Simulador implementado no Matlab.
- Implementado como sistema híbrido (Supervisor escalona controladores).

# Desenvolvimento

## Simiam

- Simulador implementado no Matlab.
- Implementado como sistema híbrido (Supervisor escala controladores).



## Alterações mais importantes

- Incluir uma classe para implementar aspectos físicos do robô deste trabalho.
- Incluir suporte para controladores *fuzzy*.

# Considerações Finais

- Autonomia em arquiteturas não deliberativas.
- Estimativa de estado.
- Comparação qualitativa.

**Universidade Tecnológica Federal do Paraná**  
Engenharia de Computação  
*Campus Pato Branco*

**Trabalho de Conclusão de Curso - 2019/1**  
Navegação de robôs móveis utilizando Lógica *Fuzzy*

Profa. Orientadora: Dra. Kathya Silvia Collazos Linares  
Acadêmico: Marcelo Gervazoni Carbonera

2 de Julho de 2019