

---

# Sistema de Controlo para Veículo Autónomo Terrestre em Ambiente Semi-estruturado

---

António Sérgio Gonçalves

Marco de Oliveira Marques

2006

---

# Trabalho realizado no âmbito da disciplina de Laboratório de Sistemas

---

## Colaboração com:

LSA, Laboratório de Sistemas Autónomos – Projecto Runner



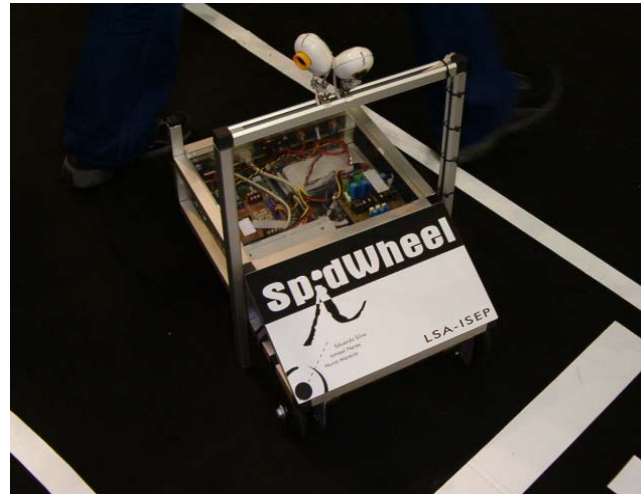
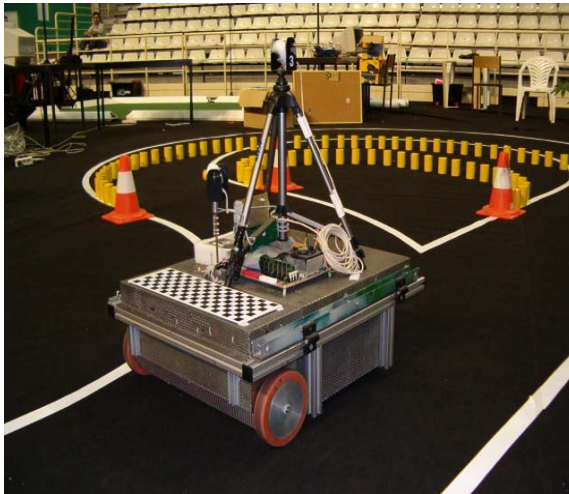
# Descrição do Trabalho

- Pretende-se dotar os veículos autónomos terrestres participantes na prova de condução do FNR de manobras de controlo de locomoção em ambientes pouco estruturados como por exemplo a condução do veículo por entre mecos de sinalização;
- O sistema de controlo deve permitir disponibilizar as várias manobras necessárias para a execução da prova.



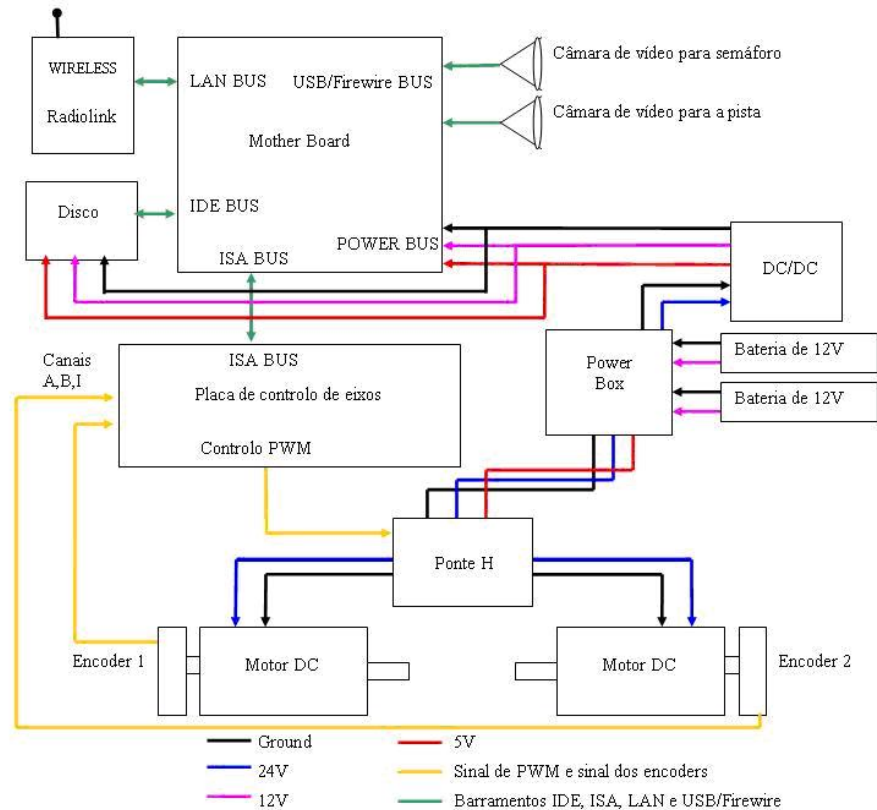
# O Veículo

- Existem dois robôs no projecto Runner;
- Dimensões limitadas pela largura da pista;
- Todas as decisões são tomadas pelos sistemas nele incluídos;
- Possuem todos os dispositivos de armazenamento de energia.



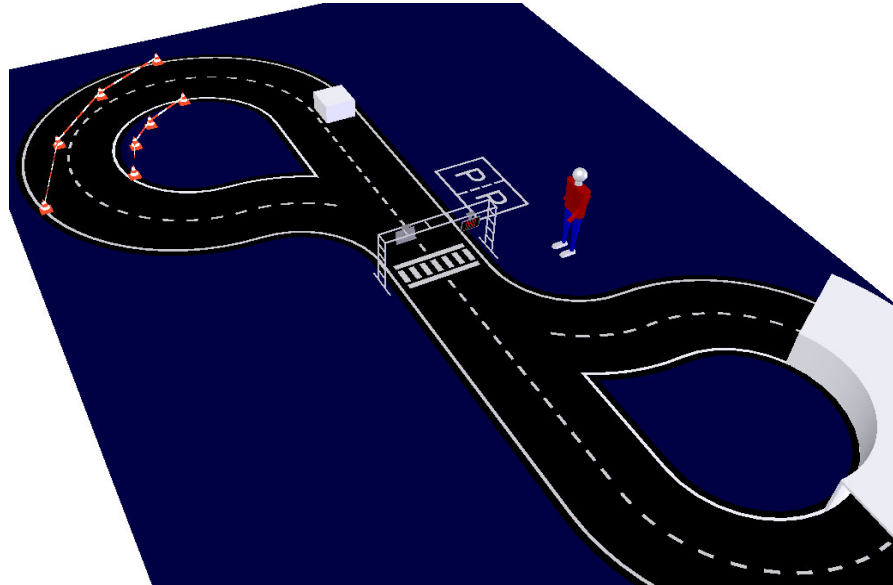
# O Veículo

## ■ Diagrama do hardware



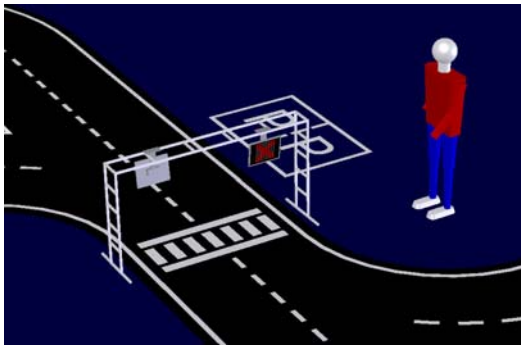
# A Prova

- A Pista
  - Área de 11x16.4m;
  - Possui o formato de estrada;
  - É delimitada por duas linhas laterais e paralelas;
  - Possui duas zonas para estacionamento.

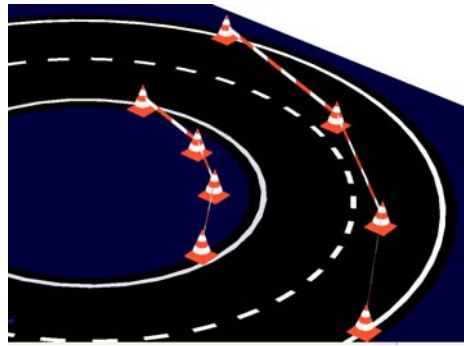


# A Prova

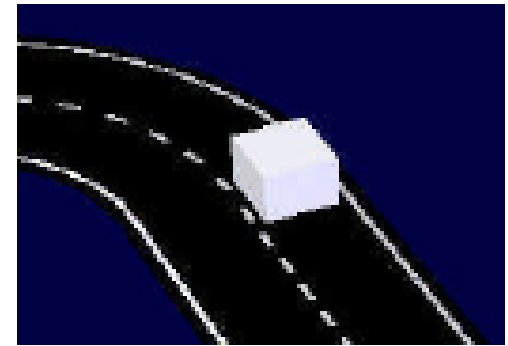
- Está dividida em três rondas, cada uma com duas passagens;
- O nível de dificuldade vai aumentando com as rondas:
  - Sem obstáculos;
  - Adicionam-se os semáforos, os obstáculos e o parque;
  - Adicionam-se o túnel e a zona de obras.



**Semáforos**



**Zona de Obras**



**Obstáculo**

# Manobras

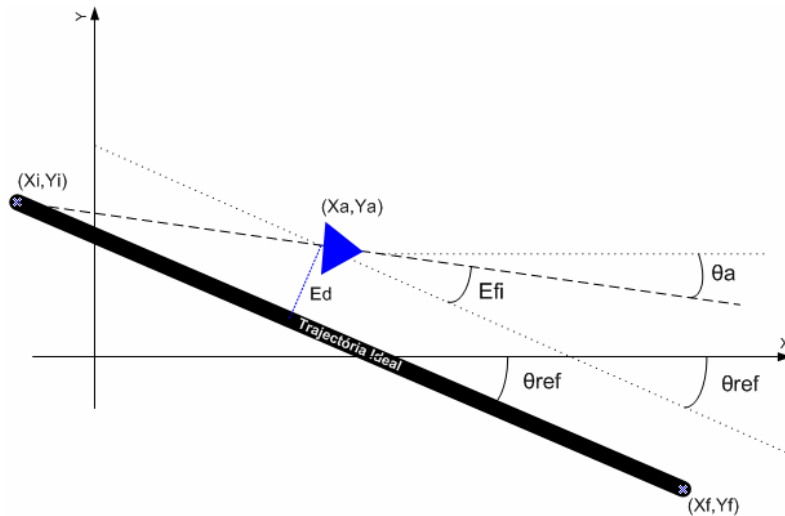
- Durante a prova o robô poderá ter de efectuar várias manobras:
  - Recta;
  - Curva;
  - Desvio de obstáculo;
  - Parar;
  - Parque;
  - Passagem por zona de obras.
  
- Existência de erros na trajectória
  - Erro de distancia à trajectória (erro em  $d$ );
  - Erro de orientação (erro em  $\theta$ ).



# Manobras

## ■ Recta

- Definida por um ponto inicial  $(X_i, Y_i)$  e um ponto final  $(X_f, Y_f)$ ;
- Trajectória rectilínea segundo um ângulo de referencia.



$$\theta_{ref} = \arctan\left(\frac{Y_f - Y_i}{X_f - X_i}\right)$$

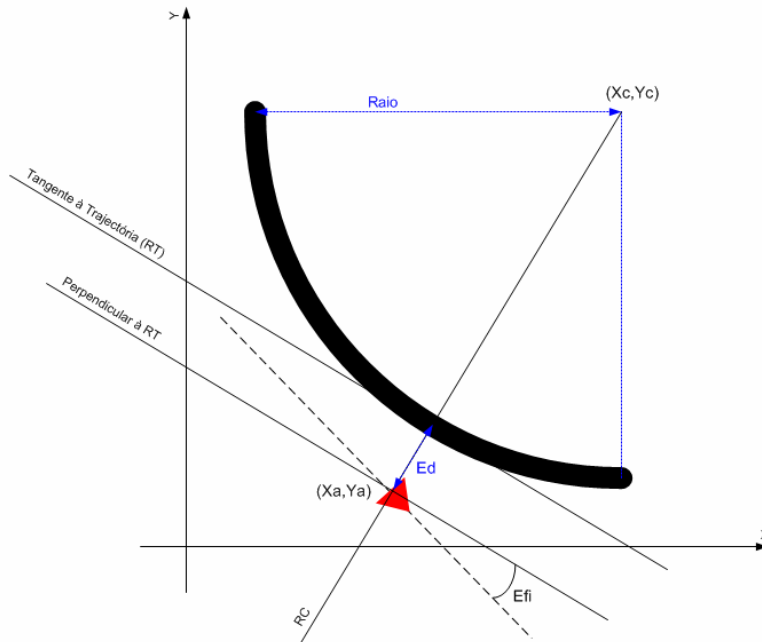
$$Efi = \theta_{REF} - \theta_{ACTUAL}$$

$$Ed = \sqrt{(X_A - X_I)^2 + (Y_A - Y_I)^2} * \sin\left(\theta_{REF} - \arctan\left(\frac{Y_A - Y_I}{X_A - X_I}\right)\right)$$

# Manobras

## ■ Curva

- ❑ Definida por um ponto central  $(X_c, Y_c)$  e um raio;
- ❑ Possibilidade de sentido horário e anti-horário;
- ❑ Orientação segundo a tangente à trajectória.



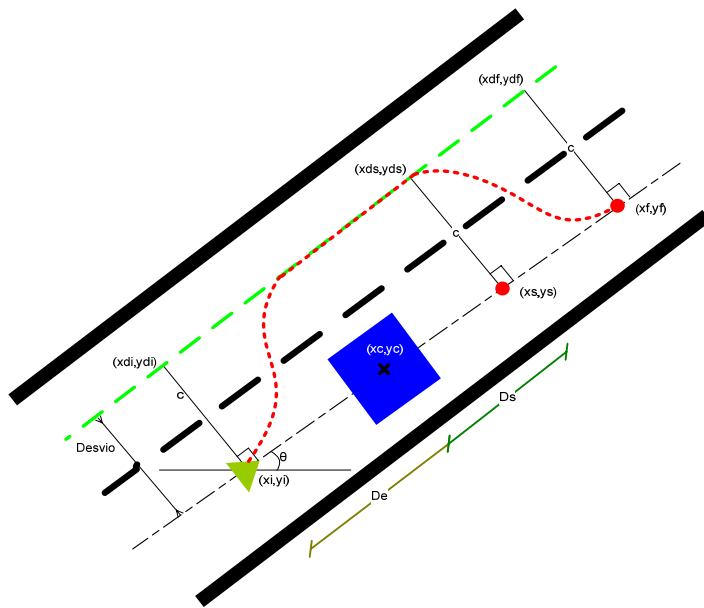
$$\theta_{ref} = \arctan\left(\frac{X_A + X_C}{Y_C - Y_A}\right)$$

$$Efi = \theta_{REF} - \theta_{ACTUAL}$$

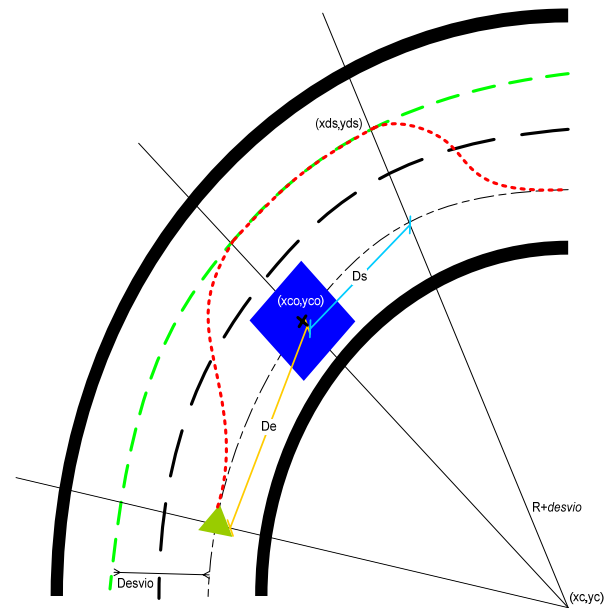
$$Ed = \sqrt{(X_C - X_A)^2 + (Y_C - Y_A)^2} - raio$$

# Manobras

- Desvio de obstáculo
  - Possibilidade de existência em recta ou curva;
  - Parametrização de uma trajectória na faixa de rodagem contrária.



Obstáculo numa recta

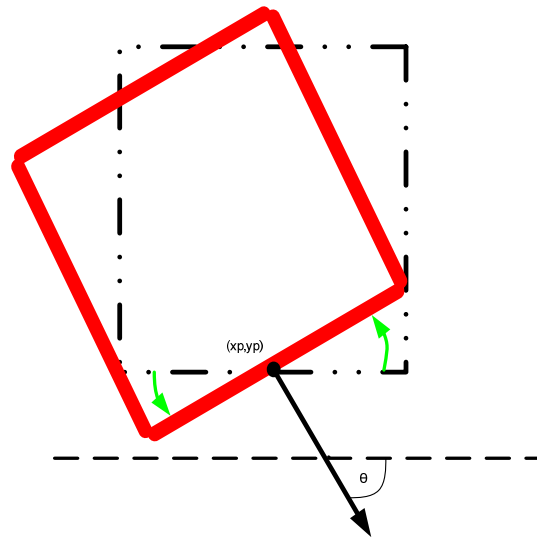


Obstáculo numa curva

# Manobras

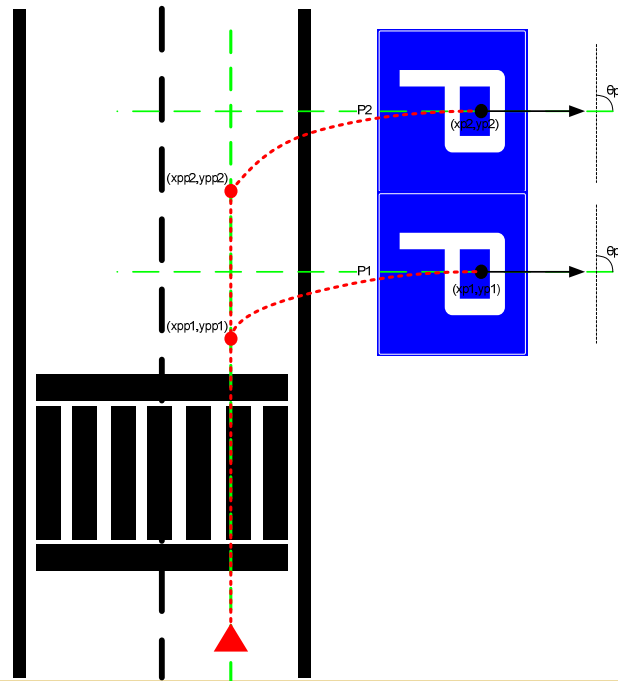
## ■ Parar

- Definida por um ponto final e uma orientação;
- Trajectória rectilínea até ao ponto final;
- Orientação depois da paragem no ponto final, através da introdução de uma velocidade diferencial.



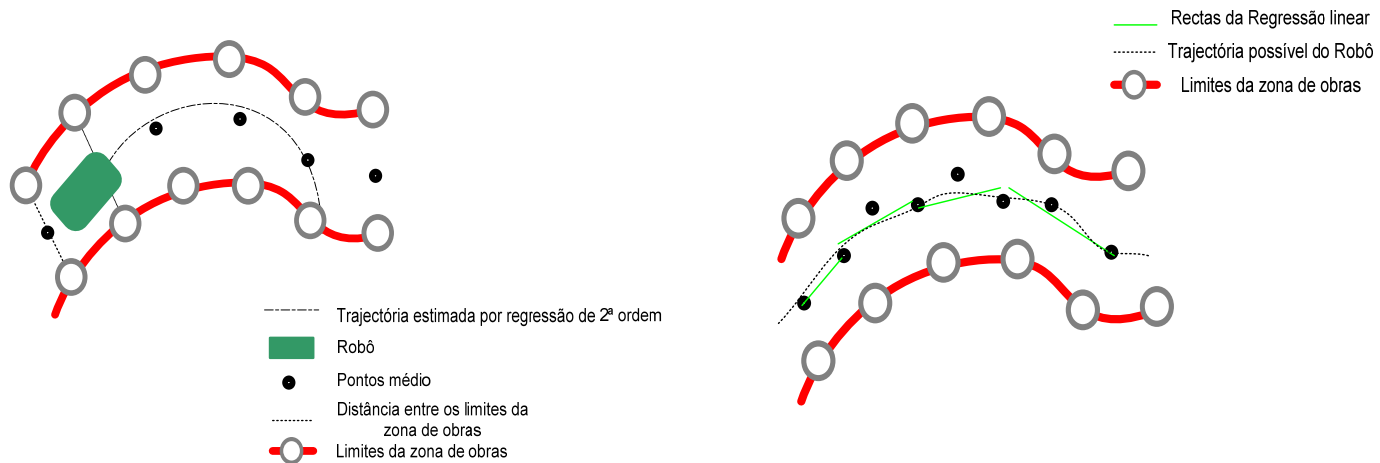
# Manobras

- Parque
  - Existem dois lugares para o estacionamento, podendo um deles estar ocupado com um obstáculo;
  - Sequencia de manobras (recta – parar).



# Manobras

- Passagem por zona de obras
  - Primeira passagem reactiva, usando a regressão quadrática;
  - Segunda passagem calculada através da regressão linear dos pontos obtidos na primeira passagem.



# Estrutura das funções

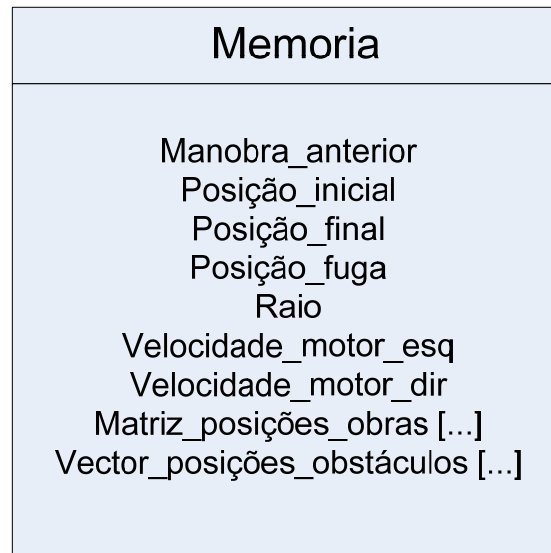
- Cada manobra recebe um contexto diferente:

RECTA	CURVA	PARAR	PARQUE
Velocidade linear Raio=0 $X_i$ - X inicial $X_f$ - X final $Y_i$ - Y inicial $Y_f$ - Y final	Velocidade linear Raio $X_c$ - X centro $Y_c$ - Y centro	Velocidade linear $X_p$ - X final $Y_p$ - Y final $\Theta$ - orientação	Velocidade linear $X_p$ - X final $Y_p$ - Y final $X_{pp}$ - X intermédio $Y_{pp}$ - Y intermédio $\Theta$ - orientação

- Manobra passagem na zona de obras o sistema de controlo tem autonomia em relação ao controlador de alto-nível, tomando todas as decisões que o permitam chegar ao final da zona de obras sem tocar nos mecos.

# Estrutura das funções

- Estrutura memória:
  - Guardar o estado actual que serão usados para o calculo do próximo estado;
  - Partilhada por todas as manobras e por outras funções (gestão de missões, etc.).





# Estrutura das funções

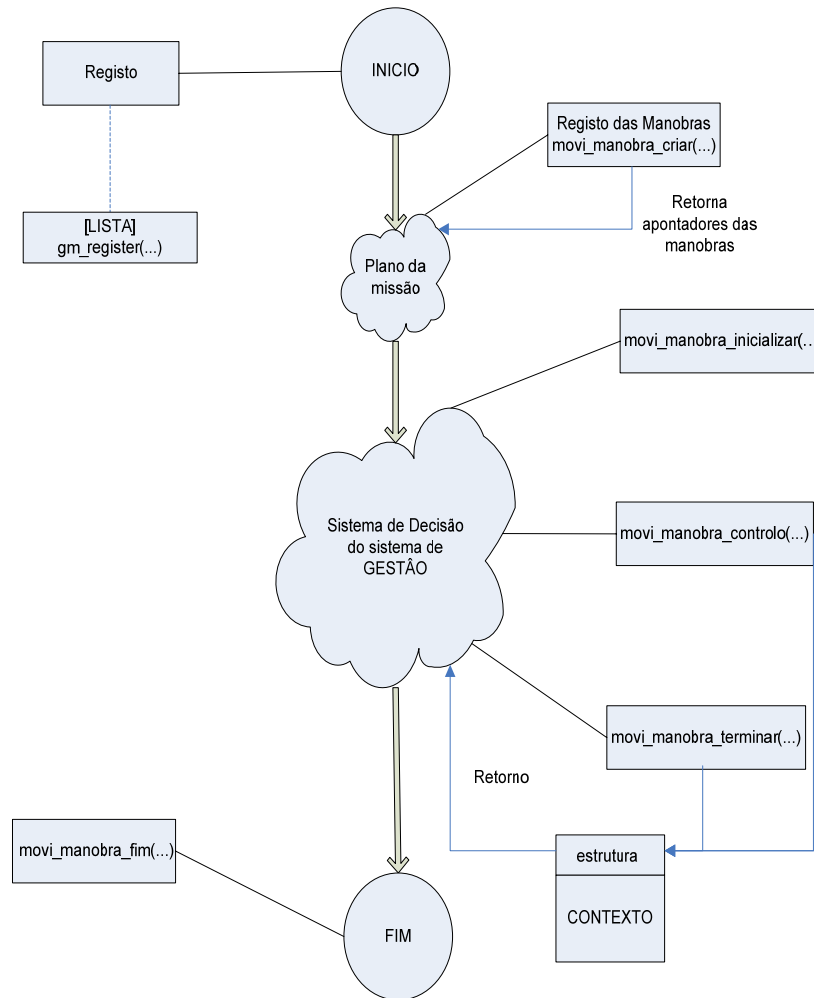
- Todas as manobras devem estar divididas em cinco funções:
  - `movi_manobra_criar(&contexto)`
    - Preparar a inicialização de cada manobra (alocar memória, criar variáveis,...).
  - `movi_manobra_inicializar(&contexto)`
    - Preparar o controlo para a manobra a executar.
  - `movi_manobra_controlo(&contexto)`
    - Controlar a trajectória do robo para a manobra pedida.
  - `movi_manobra_terminar(&contexto)`
    - Guarda as informações necessárias para os passos que se seguirão.
  - `movi_manobra_fim(&contexto)`
    - Liberta os recurso criados pela `movi_manobra_criar(&contexto)`;
    - Utilizada quando é feito o RESET ao sistema.

---

# Modelo para o controlo

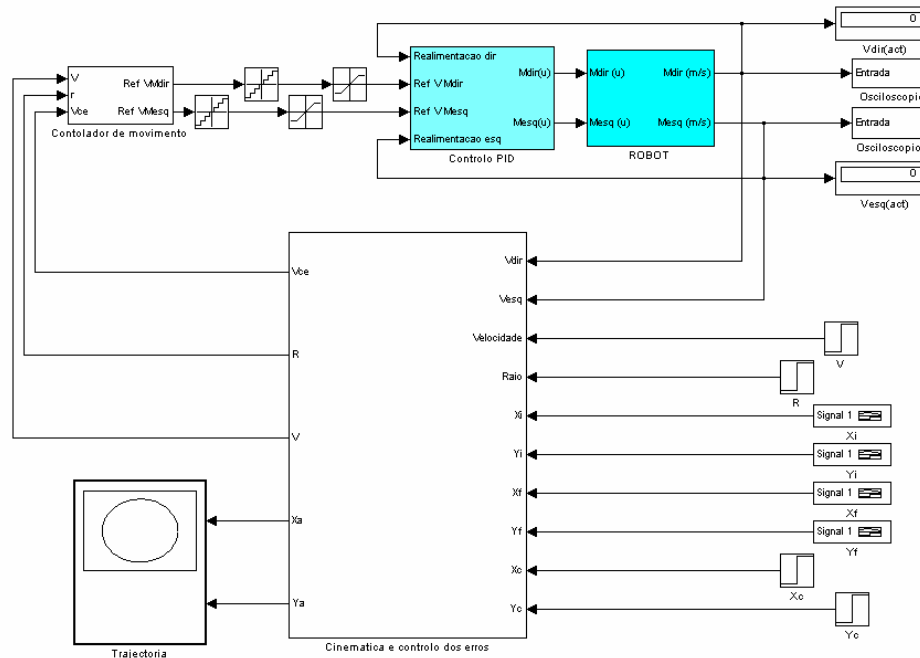
- Programa de controlo é chamado ciclicamente pelo programa de gestão;
- Certifica-se que a lei de controlo para a manobra pedida é correctamente realizada;
- Quando efectuar todos os passos anteriores deve guardar toda a informação relevante (referente ao seu estado) e “morrer”.

# Modelo para o controlo



# Controlador

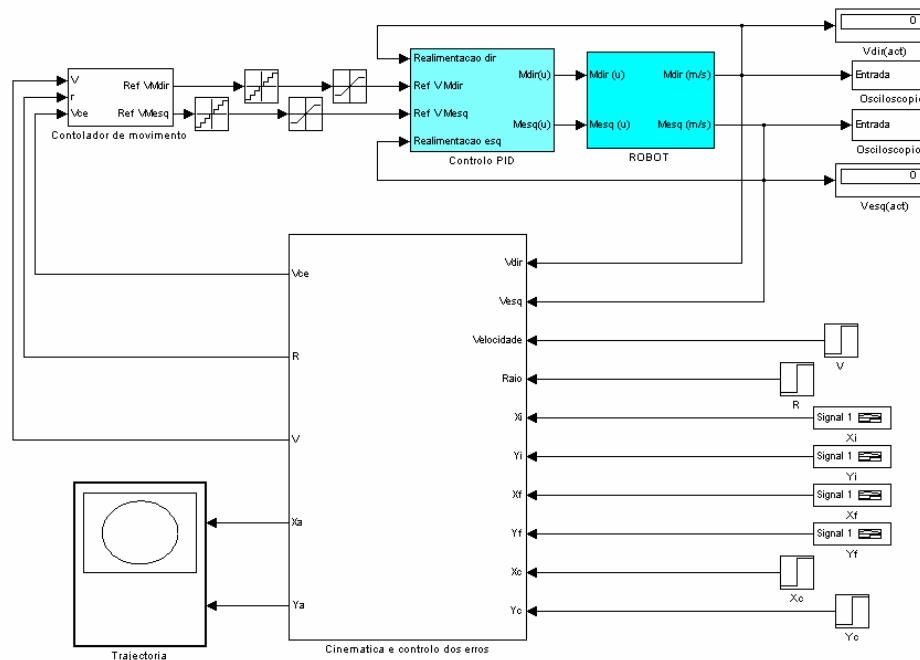
- Simulação em Simulink (matlab);
- Composto por vários blocos com diferentes funções;



Vista geral dos blocos do controlador

# Controlador

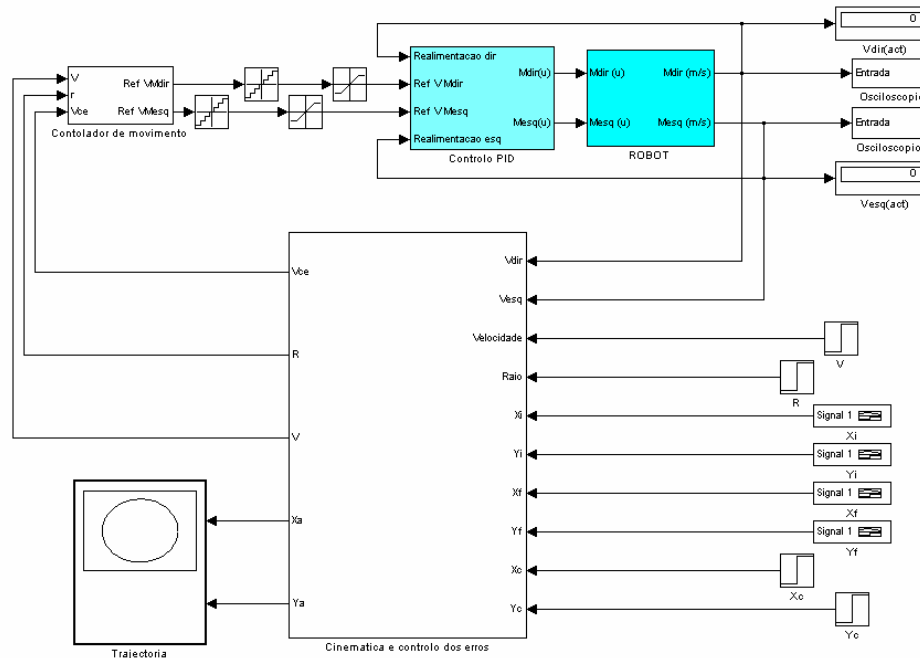
- Os blocos a azul já se encontram implementados fisicamente no robô;
- O ficheiro .mdl (matlab) do controlador encontra-se disponível em <http://lsis.no.sapo.pt>.



Vista geral dos blocos do controlador

# Controlador

- As saturações e as quantificações existentes à entrada do bloco referenciado como “Controlo PID” correspondem às limitações impostas pela ponte H e pelo microcontrolador respectivamente.

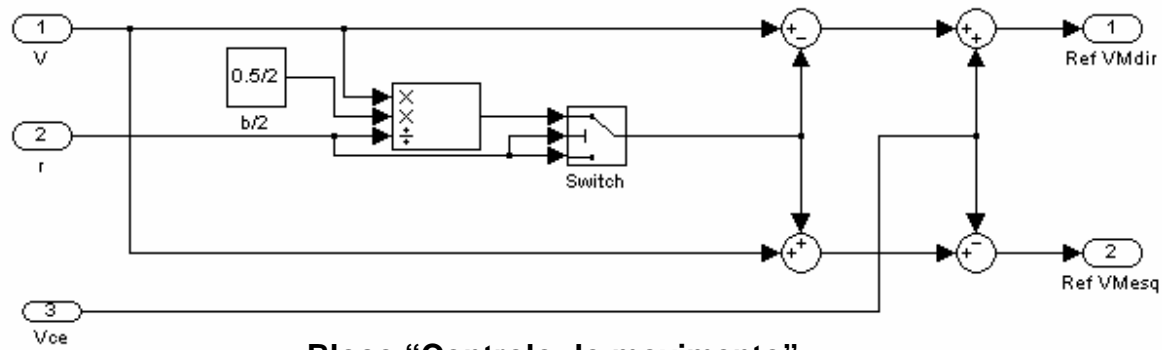


Vista geral dos blocos do controlador

# Controlador

## ■ Bloco “Controlo do movimento”

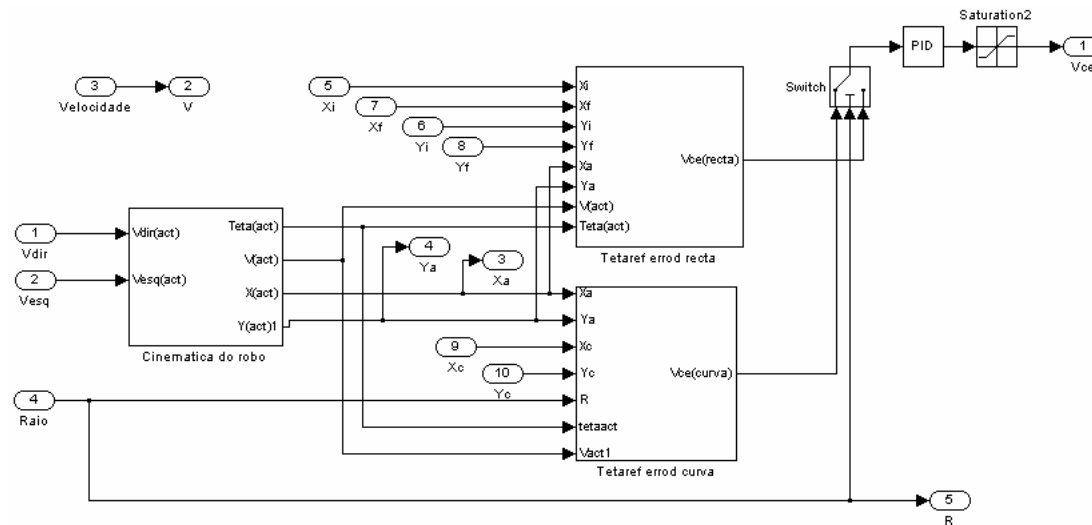
- ❑ Determina a velocidade de cada motor para que a trajectória obtida seja o mais próximo possível da trajectória desejada;
- ❑ À velocidade linear ( $V$ ) é adicionada uma velocidade diferencial, calculada com base no raio e nos erros ( $V_{ce}$ );
- ❑ Raio ( $r$ )=0  $\Rightarrow$  recta.
- ❑ Raio ( $r$ )>0  $\Rightarrow$  curva no sentido horário
- ❑ Raio ( $r$ )>0  $\Rightarrow$  curva no sentido anti-horário



Bloco “Controlo do movimento”

# Controlador

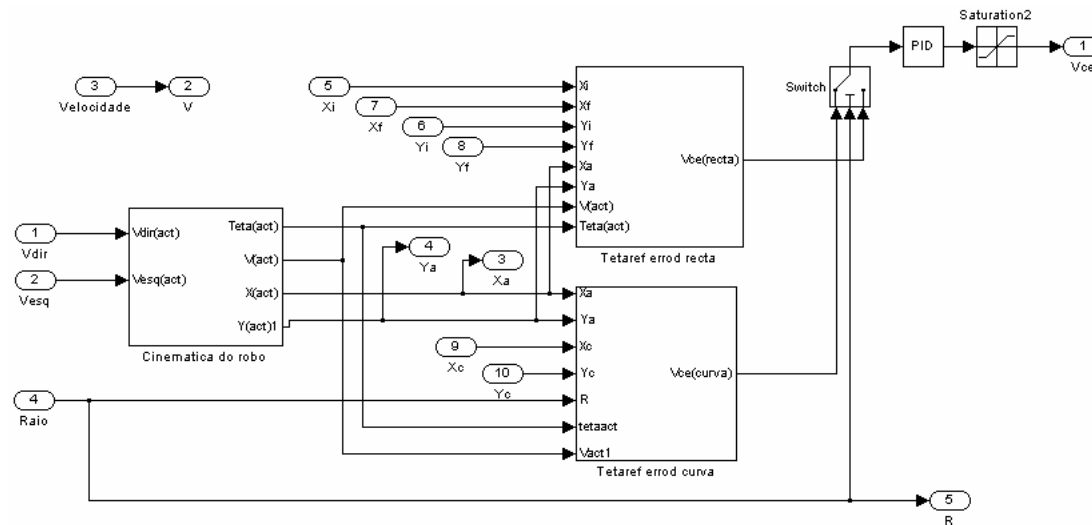
- Bloco “Cinemática e controlo de erros”
  - Bloco “Cinemática do robo” calcula a posição e a orientação actuais através da velocidade de cada um dos motores.
  - Blocos “Tetaref Errod Recta “ e “Tetaref Errod Curva“ calculam a velocidade diferencial ( $V_{ce}$ ) necessária para que sejam corrigidos os erros quer de orientação quer de distancia à trajectória.





# Controlador

- Bloco “Cinemática e controlo de erros”
  - ❑ O PID associado a  $V_{ce}$  permite uma antecipação na correcção do erro;
  - ❑ A saturação deverá ter como valor máximo a velocidade linear actual;
  - ❑ O switch determina em função do raio qual o bloco que deve calcular a  $V_{ce}$ .

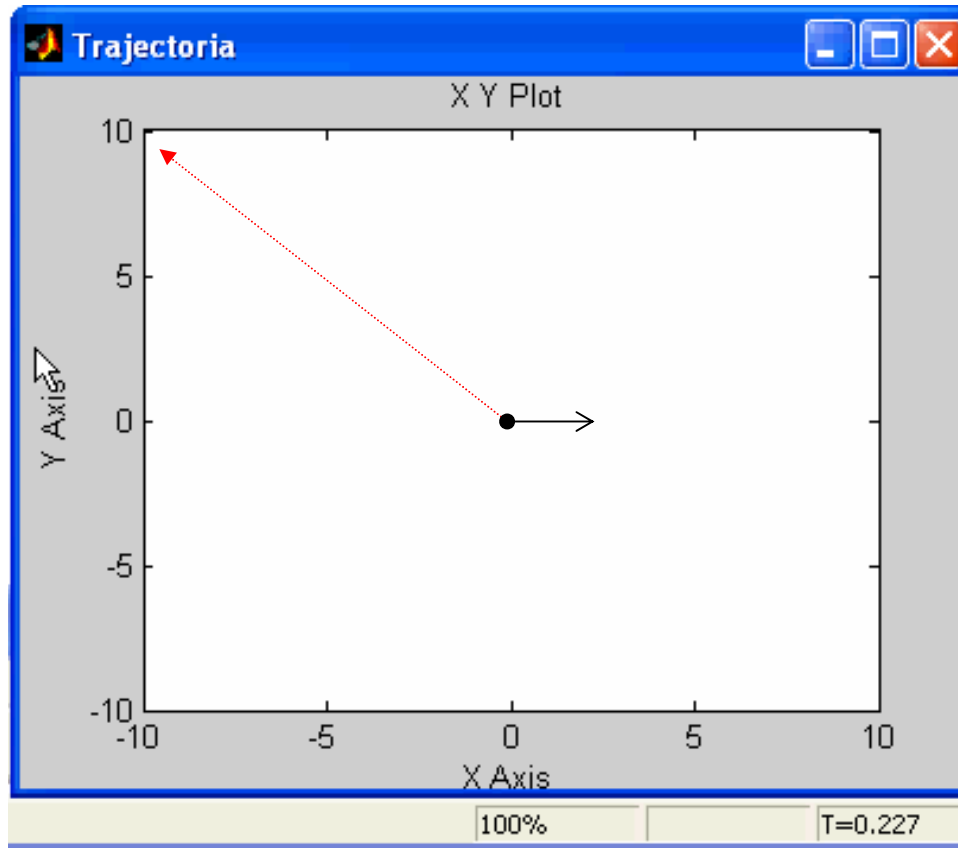


# Exemplo do funcionamento do controlador

- Inicialmente o robô encontra-se na posição  $(0,0)$ , com ângulo de  $0^\circ$ .
- Sequencia de manobras:
  - $t=0$  - recta com origem em  $(0,0)$  e que passa no ponto  $(-1,1)$ ;
  - $t=5$  - recta com origem em  $(0,0)$  e que passa no ponto  $(-1,-1)$ ;
  - $t=15$  - curva com centro em  $(0,0)$  e raio 4 no sentido horário;
  - $t=30$  - curva com centro em  $(0,4)$  e raio 4 no sentido horário;
  - $t=60$  - curva com centro em  $(4,4)$  e raio 4 no sentido horário;

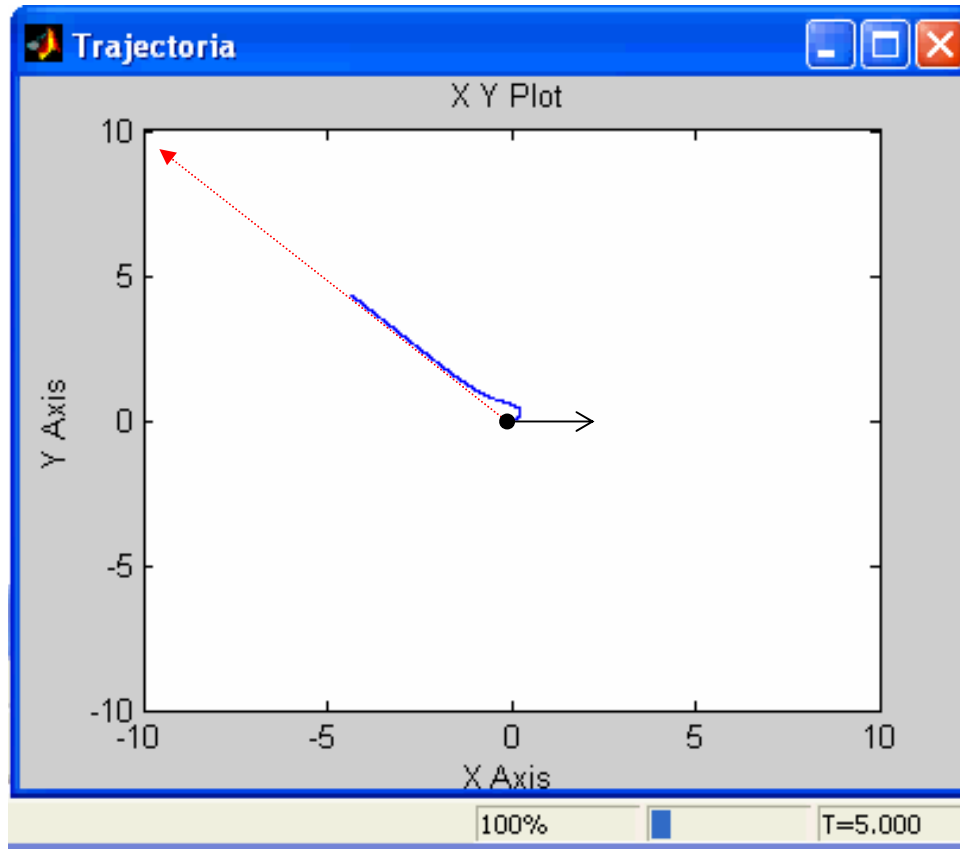
# Exemplo do funcionamento do controlador

- $t=0$  - recta com origem em  $(0,0)$  e que passa no ponto  $(-1,1)$ ;



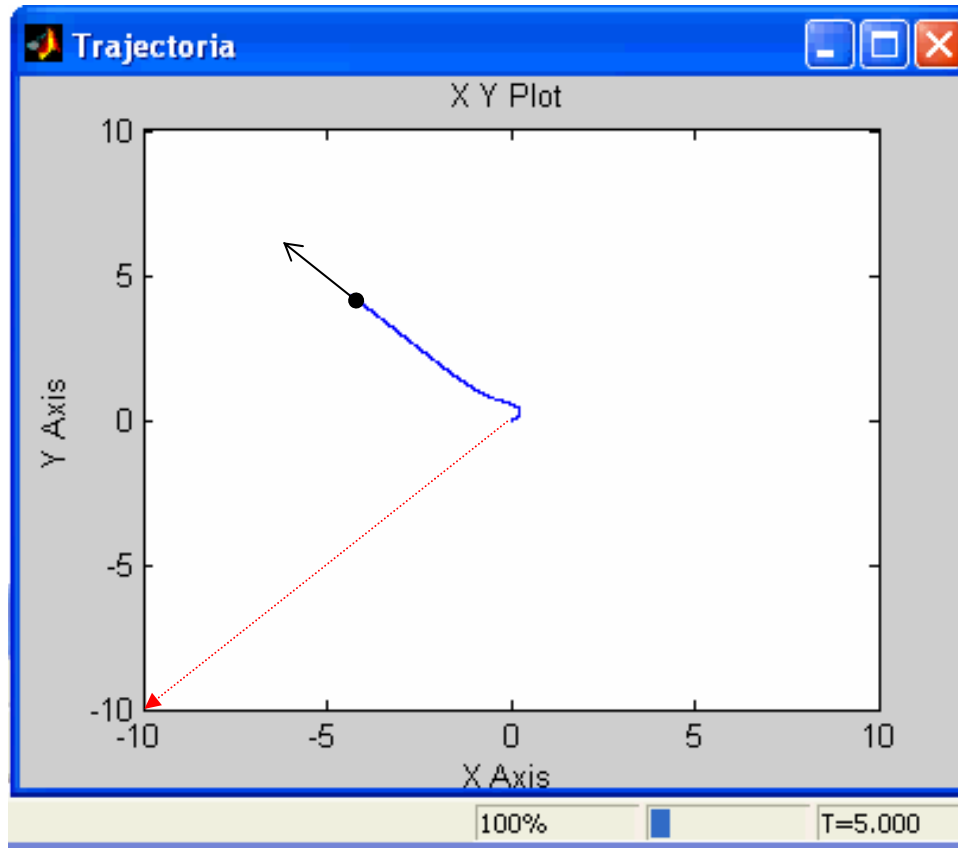
# Exemplo do funcionamento do controlador

- $t=0$  - recta com origem em  $(0,0)$  e que passa no ponto  $(-1,1)$ ;



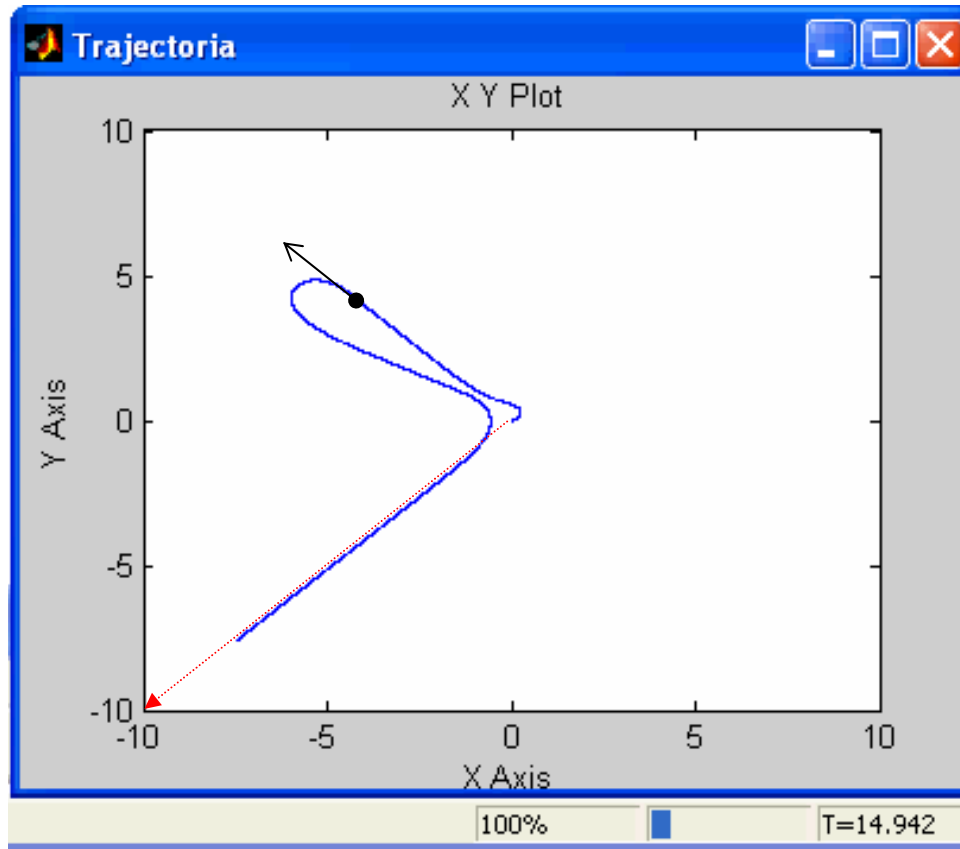
# Exemplo do funcionamento do controlador

- $t=5$  - recta com origem em  $(0,0)$  e que passa no ponto  $(-1,-1)$ ;



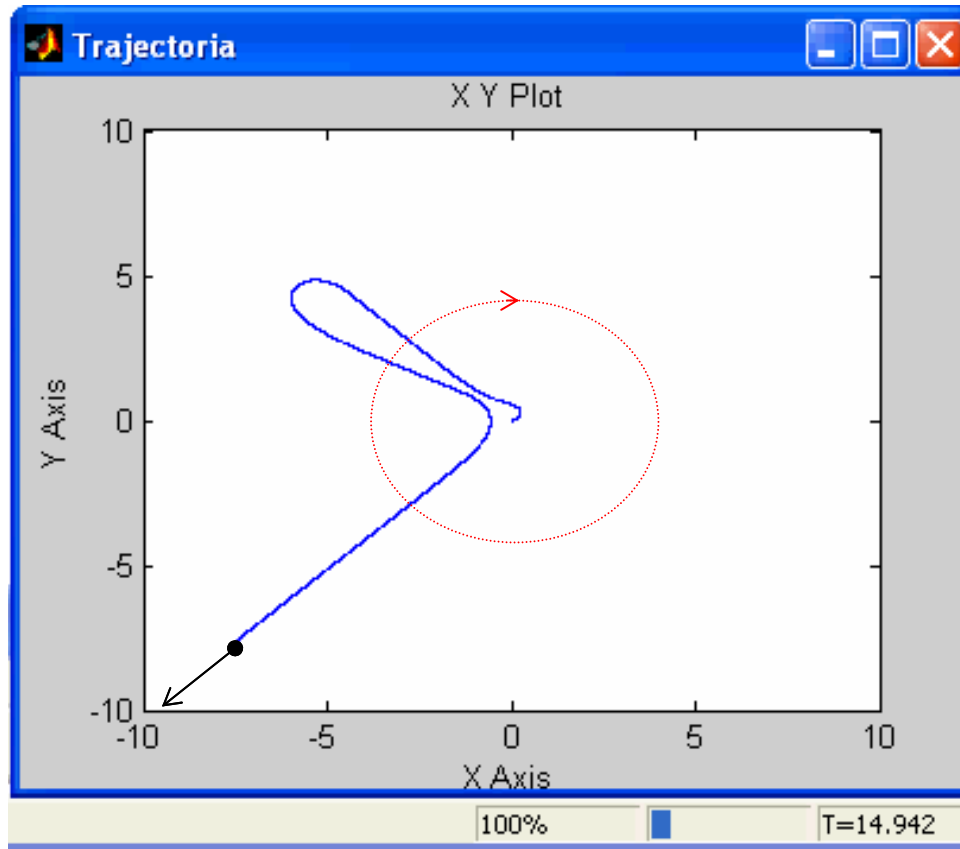
# Exemplo do funcionamento do controlador

- $t=5$  - recta com origem em  $(0,0)$  e que passa no ponto  $(-1,-1)$ ;



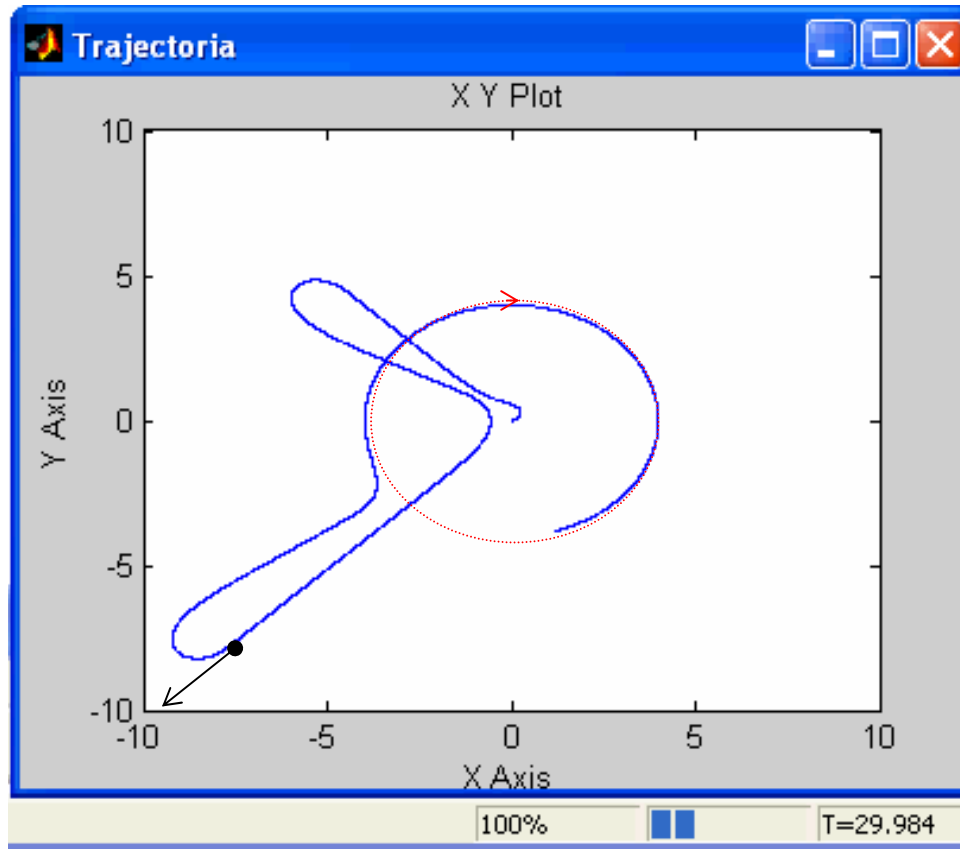
# Exemplo do funcionamento do controlador

- $t=15$  - curva com centro em  $(0,0)$  e raio 4 no sentido horário;



# Exemplo do funcionamento do controlador

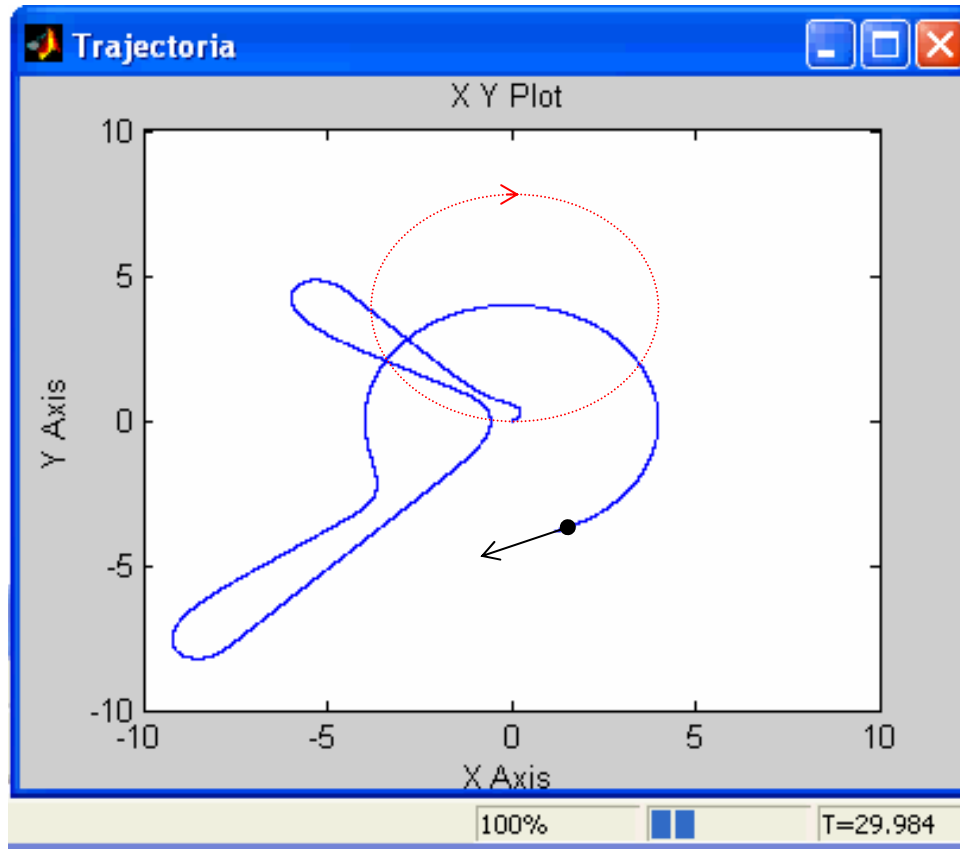
- $t=15$  - curva com centro em  $(0,0)$  e raio 4 no sentido horário;





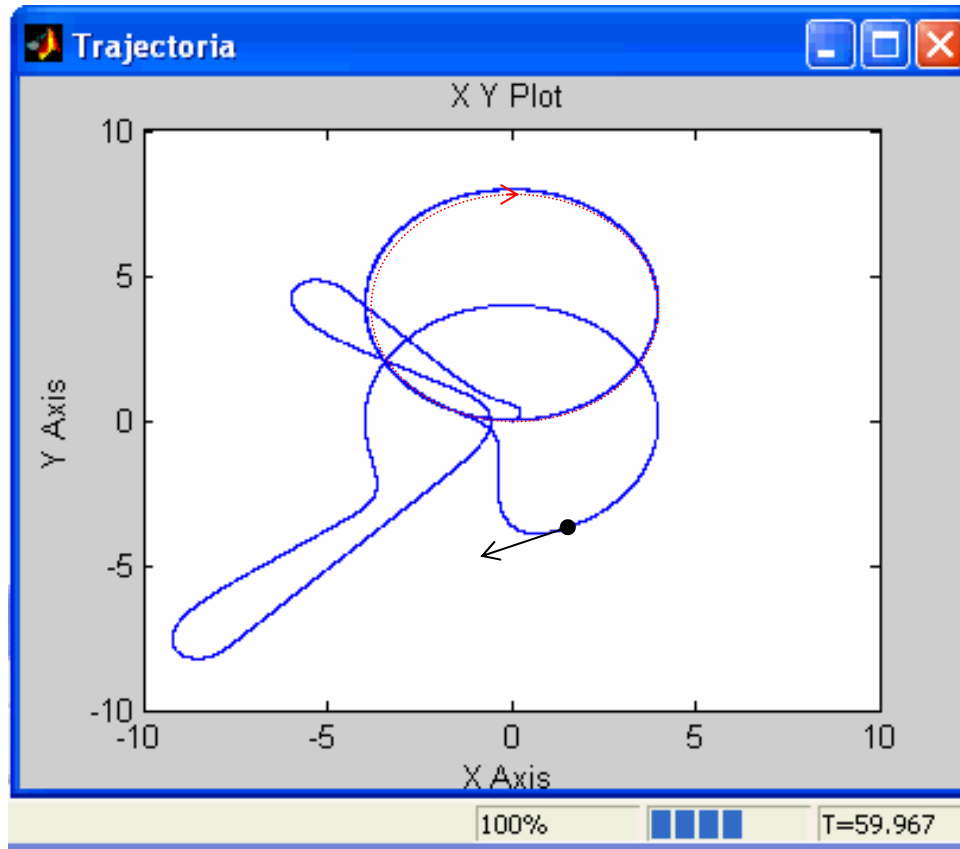
# Exemplo do funcionamento do controlador

- $t=30$  - curva com centro em  $(0,4)$  e raio 4 no sentido horário;



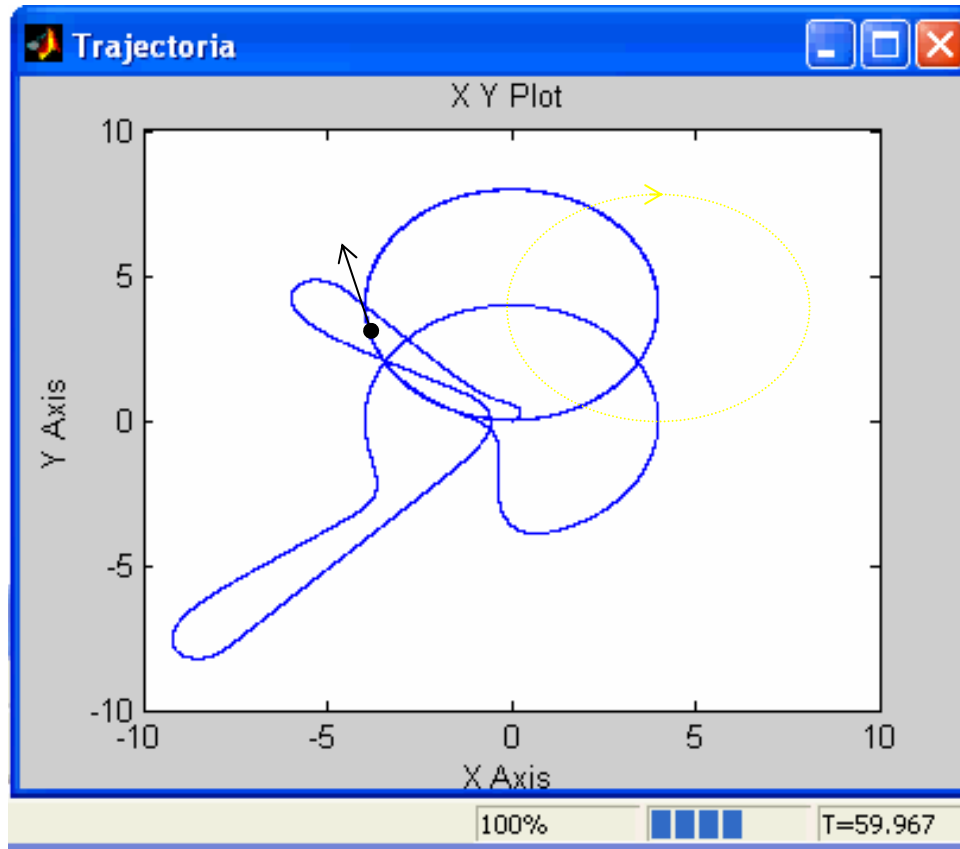
# Exemplo do funcionamento do controlador

- $t=30$  - curva com centro em  $(0,4)$  e raio 4 no sentido horário;



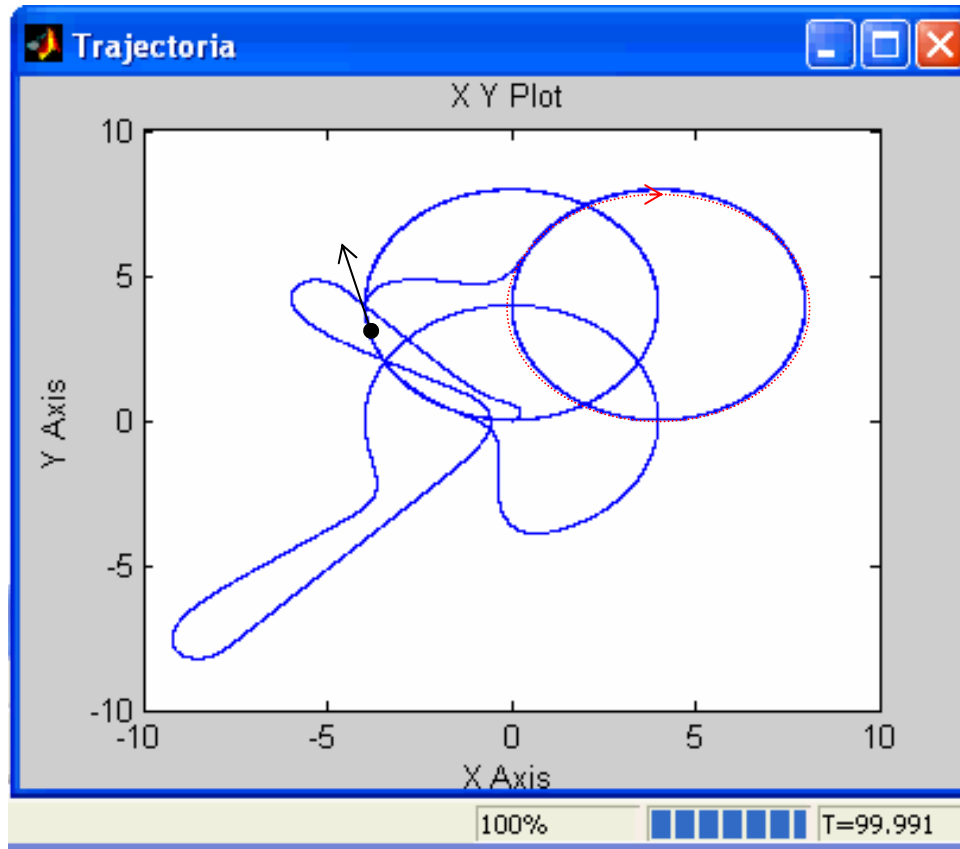
# Exemplo do funcionamento do controlador

- $t=60$  - curva com centro em  $(4,4)$  e raio 4 no sentido horário;



# Exemplo do funcionamento do controlador

- $t=60$  - curva com centro em  $(4,4)$  e raio 4 no sentido horário;



---

FIM

---

Obrigado pela vossa atenção