



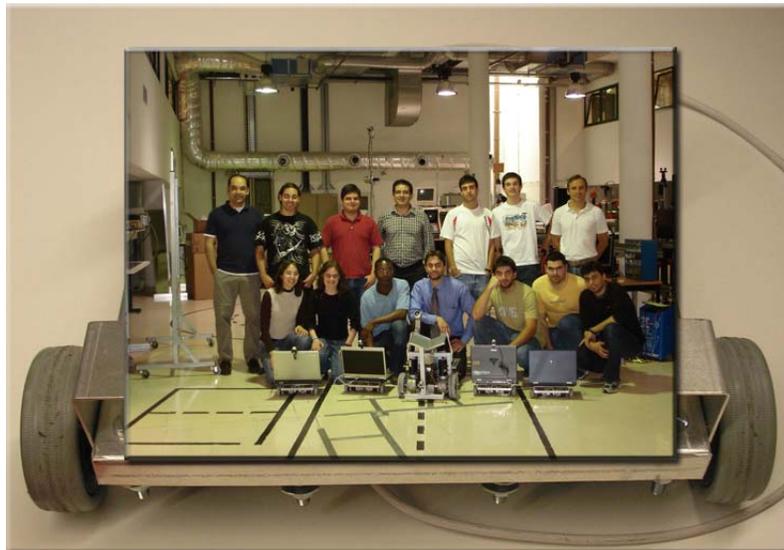
INSTITUTO SUPERIOR TÉCNICO

Licenciatura em Engenharia Mecânica

# CONTROLO DE SISTEMAS

## COMPETIÇÃO de RASTEIRINHOS

*14 de Junho de 2007*



**Patrícia Ferreira 55498**  
**Carlos Costa 55518**  
**Ana Mestre 55554**  
**3ºAno**

IST, 2006 – 2007

# Índice

<b>ÍNDICE .....</b>	<b>2</b>
<b>1 - OBJECTIVOS .....</b>	<b>3</b>
<b>2 - BASES TEÓRICAS .....</b>	<b>4</b>
2.1 – IDENTIFICAÇÃO DO SISTEMA.....	4
2.1.1 – <i>Identificação da Função de Transferência</i> .....	4
2.1.2 – <i>Análise do comportamento do sistema através do LGR</i> .....	5
2.2 – CONTROLO .....	6
2.2.1 – <i>Controlador PI</i> .....	6
2.2.2 – <i>Controlador PD</i> .....	7
2.2.3 – <i>Controlador PID</i> .....	8
2.2.4 – <i>Compensador de Avanço</i> .....	8
2.2.5 – <i>Compensador de Atraso</i> .....	9
2.3 – MÉTODOS PARA O PROJECTO DE CONTROLADORES .....	9
2.3.1 – <i>Via LGR</i> .....	9
2.3.2 – <i>Ziegler-Nichols</i> .....	10
<b>3 - RASTEIRINHO.....</b>	<b>11</b>
<b>4 - PASSOS PARA A EXECUÇÃO DO PROJECTO .....</b>	<b>12</b>
4.1 – PASSOS INICIAIS .....	12
4.2 – FUNÇÃO DE TRANSFERÊNCIA PARA O CONTROLO DE POSIÇÃO .....	12
4.3 – PROJECTO DO CONTROLADOR .....	13
4.4 – CONTROLO DE VELOCIDADE.....	13
<b>5 - RESULTADOS OBTIDOS .....</b>	<b>16</b>
<b>6 - COMPARAÇÃO DE RESULTADOS ENTRE CONTROLADORES .....</b>	<b>20</b>
6.1 – PID .....	20
6.2 – PI.....	21
6.3 – PD .....	21
6.4 – COMPENSADOR DE AVANÇO.....	22
6.5 – COMPENSADOR DE ATRASO.....	23
<b>7 - RESULTADOS OBTIDOS NA COMPETIÇÃO.....</b>	<b>24</b>
<b>8 - DISCUSSÃO DE RESULTADOS .....</b>	<b>26</b>
<b>9 - PONTOS A TER EM CONTA PARA A EXECUÇÃO DO PROJECTO .....</b>	<b>29</b>
<b>10 - CONCLUSÕES .....</b>	<b>33</b>
<b>11 - BIBLIOGRAFIA .....</b>	<b>34</b>
<b>12 - AGRADECIMENTOS.....</b>	<b>35</b>

# 1 - Objectivos

O projecto em estudo teve como objectivo controlar o robot Rasteirinho, utilizando os conhecimentos adquiridos na cadeira de Controlo de Sistemas, fazendo-o seguir uma linha preta a tracejado. Pretendeu-se numa fase posterior fazer o controlo da velocidade do Rasteirinho. Era como objectivo fazer este controlo através de um algoritmo de contagem dos traços da linha, mas como tal não foi possível o controlo de velocidade foi feito utilizando um método de Controlo Híbrido.

Neste trabalho utilizou-se os recursos disponíveis no software Matlab, bastante úteis neste tipo de projectos, como o *Simulink* onde foi implementado o sistema de controlo. O ficheiro inicial foi disponibilizado por colegas, visto não ser objectivo deste trabalho a sua implementação de raiz.

## 2 - Bases Teóricas

Na elaboração deste projecto há que ter bem assente algumas noções de temas tratados na cadeira de Controlo de Sistemas. É assim essencial saber identificar uma Função de Transferência via Resposta no Tempo, saber analisar o LGR para a análise da estabilidade e do comportamento do sistema e saber as características de cada controlador que poderia vir a ser implementado.

### 2.1 – Identificação do sistema

Para se controlar um sistema há que primeiro identificá-lo. Nos sistemas reais, como o Rasteirinho, não se tem conhecimento *a priori* da sua Função de Transferência (FT), como tal, há que fazer experiências para a determinar. Essas experiências baseiam-se no estudo do comportamento do sistema utilizando um método para a identificação do mesmo, existindo para isso duas hipóteses: Estudo através da Resposta em Frequência ou via Resposta no Tempo.

Entende-se por Resposta em Frequência como sendo o mapeamento dos pontos singulares da Função de Transferência num ponto do eixo imaginário, de zero a mais infinito, retirando-se em cada ponto os valores da fase e da magnitude para a construção do Diagrama de Bode.

A análise via Resposta no Tempo consiste na observação de um gráfico que represente a resposta do sistema quando sujeita a solicitações, por exemplo, a um degrau, e retirar dele os parâmetros que permitam chegar à Função de Transferência.

O método utilizado neste trabalho foi via Resposta no Tempo, como tal, apresenta-se aqui em maior detalhe os procedimentos para a obtenção da FT pretendida.

#### 2.1.1 – Identificação da Função de Transferência

Obtidas as localizações dos pólos, se estes forem complexos conjugados consegue-se determinar a Função de Transferência do anel fechado, sabendo que:

$$G(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (1)$$

$$p_{1,2} = \frac{-2\xi\omega_n \pm \sqrt{4\xi^2\omega_n^2 - 4\omega_n^2}}{2} = -\xi\omega_n \pm j\omega_n\sqrt{1-\xi^2} \quad (2)$$

Mas como a Função de Transferência do Rasteirinho que se pretende controlar corresponde à Função de Transferência em anel aberto, há que a determinar. Facilmente

se demonstra através da simplificação do Diagrama de Blocos que ambas se relacionam por:

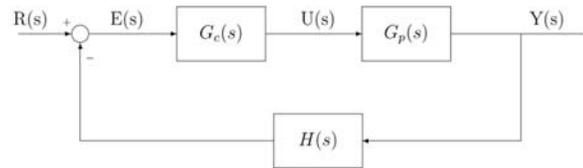


Fig. 1 – Diagrama de Blocos

$$\frac{Y(s)}{R(s)} = \frac{G_c(s)G_p(s)}{1 + G_c(s)G_p(s)H(s)} \quad (3)$$

Onde a Função de Transferência do anel aberto, em que  $G(s)$  é uma FT de ganho unitário, é:

$$G_c(s)G_p(s)H(s) = KG(s) \quad (4)$$

De onde resulta a FT do anel fechado:

$$\frac{Y(s)}{R(s)} = \frac{G_c(s)G_p(s)}{1 + KG(s)} \quad (5)$$

### 2.1.2 – Análise do comportamento do sistema através do LGR

O Lugar Geométrico das Raízes (LGR) não é mais do que um método gráfico que representa o “caminho” que os pólos da FT do anel fechado fazem à medida que se varia unicamente o ganho da FT do anel aberto. Este método permite assim verificar rapidamente as características principais do sistema em anel fechado, através da sua FT em anel aberto, como por exemplo a sua estabilidade.

Resumidamente, para a análise visual do LGR sabe-se que:

- Aumento do módulo da parte real dos pólos dominantes (pólos mais afastados do eixo imaginário):
  - O tempo de estabelecimento diminui o que resulta numa resposta mais rápida;
  - O máximo de Sobreimpulso diminui;
  - A frequência natural amortecida e o tempo de subida mantêm-se.

- Aumento da parte imaginária:
  - O Sobreimpulso e a frequência natural amortecida aumentam, tornando a resposta mais oscilatória;
  - O tempo de subida diminui;
  - O tempo de estabelecimento mantém-se.

Uma análise semelhante pode ser feita tendo em conta os parâmetros  $\xi$  e  $\omega_n$  :

- Aumentando o  $\xi$ , mantendo  $\omega_n$  constante:
  - O máximo de Sobreimpulso diminui, assim como a frequência natural amortecida e o tempo de estabelecimento.
- Aumento do  $\omega_n$ , mantendo  $\xi$  constante:
  - O máximo de Sobreimpulso mantém-se;
  - O tempo de subida e de estabelecimento diminuem.

## **2.2 – Controlo**

Identificada a FT do anel aberto, pode-se passar à fase de controlo. O papel do controlador num anel de realimentação é o de alterar a resposta natural do sistema para que este, quando sujeito a perturbações ou variações na referência, seja capaz de as ultrapassar e se torne insensível a estas.

Dependendo da FT obtida, da localização dos pólos/zeros da mesma e do que se pretende do sistema, há que escolher o controlador que melhor se aplica.

Existem inúmeros controladores que podem ser determinados, dependendo da complexidade da FT em estudo, não tendo de ser necessariamente um controlador já existente. Mas tendo em conta os conhecimentos adquiridos na disciplina de Controlo de Sistemas, há que verificar qual o melhor controlador dos cinco apresentados em seguida, conhecendo para isso as propriedades de cada um.

Cada controlador tem por base as acções de controlo Proporcional, Integral e/ou Derivativa que podem ser usadas individualmente (excepto a componente derivativa como se verá mais à frente). No entanto, normalmente, apenas um ganho não é suficiente para controlar o sistema e, como tal, recorre-se à junção das acções de controlo P, I e D dependendo do que se pretende obter dando assim origem aos Controladores Convencionais PI, PD e PID. Há ainda mais dois controladores que poderiam ser utilizados neste projecto, o Compensador de Avanço e o Compensador de Atraso.

Apresenta-se agora, mais concretamente, as características de cada controlador.

### **2.2.1 – Controlador PI**

Este controlador é utilizado no caso particular de se precisar de eliminar o Erro Estacionário.

A acção de controlo Integral, presente neste controlador, faz com que haja uma degradação das condições do regime transitório e uma diminuição da estabilidade

relativa do sistema, podendo também provocar o “fenómeno” *reset-windup* que consiste na existência de oscilações indesejáveis na presença de saturações na acção de controlo.

A expressão temporal do controlador PI é:

$$u(t) = K_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(\eta) d\eta \right] \quad (6)$$

Resultando após Transformada de Laplace em:

$$\frac{U(s)}{E(s)} = K_p \left( \frac{T_i s + 1}{T_i s} \right) \quad (7)$$

Onde a  $T_i$  se dá o nome de Constante de Integração e  $K_p$  o Ganho Proporcional.

Pela análise da expressão anterior, verifica-se que a consequência da utilização deste controlador é a colocação de um zero real, no Semi-Plano Esquerdo (SPE), e um pólo na origem. Desta maneira, quanto maior a Constante de Integração,  $T_i$ , mais perto está o zero da origem e menor o efeito do integral do Erro. Quanto ao aumento de  $K_p$ , este faz com que o sistema fique mais rápido a responder mas também mais oscilatório.

### 2.2.2 – Controlador PD

Devido à não existência da componente Integral, este controlador não elimina o Erro Estacionário, podendo no entanto diminuir-lo. Assim, este controlador deve ser utilizado quando se pretende uma maior rapidez de resposta e a questão do Erro Estacionário está resolvida.

A acção Derivativa tem como principal característica favorável o aumento da estabilidade relativa do anel de controlo, fazendo-se assim sentir o seu efeito no regime transiente. No entanto tem a desvantagem de amplificar o ruído e por isso nunca se deve utilizar apenas a acção de controlo D. Deve-se ter também cuidado no uso do PD com o *Derivative Kick*. Este fenómeno acontece quando há uma variação brusca na referência. Ou seja, como inicialmente a diferença entre o Erro e a referência passa de zero até ao seu valor máximo, o termo derivativo vai consequentemente ser muito elevado, provocando esse efeito indesejado. Assim, a solução é colocar a acção derivativa no anel de realimentação passando a denominar-se P-D.

O controlador PD tem por base as seguintes expressões:

$$u(t) = K_p \left[ e(t) + T_d \frac{de(t)}{dt} \right] \quad (8)$$

Aplicando T.L.

$$\frac{U(s)}{E(s)} = K_p (1 + T_d s) \quad (9)$$

Onde  $K_p$  já foi referido como sendo o Ganho Proporcional e  $T_d$  corresponde à Constante Derivativa.

O que na prática significa colocar um zero no SPE e “puxar” o LGR para a esquerda, introduzindo assim avanço de fase no anel aberto. A variação dos seus parâmetros faz com que a resposta do sistema varie da seguinte maneira: O aumento do parâmetro  $T_d$  origina uma resposta mais lenta, enquanto que o aumento de  $K_p$  diminui o Erro Estacionário.

### 2.2.3 – Controlador PID

O PID é uma junção dos dois controladores já apresentados e por isso o mais utilizado pois tenta compensar os aspectos indesejáveis de uma acção de controlo com uma outra. Isto é, para eliminar o Erro Estacionário utiliza-se a componente Integral e para controlar as oscilações indesejáveis da resposta transitória, resultantes da componente I, utiliza-se a componente Derivativa.

A resposta temporal é dada pela soma das componentes do PI e do PD:

$$u(t) = K_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(\eta) d\eta + T_d \frac{de(t)}{dt} \right] \quad (10)$$

Resultando na Função de Transferência,

$$\frac{U(s)}{E(s)} = K_p \left( \frac{T_i T_d s^2 + T_i s + 1}{T_i s} \right) \quad (11)$$

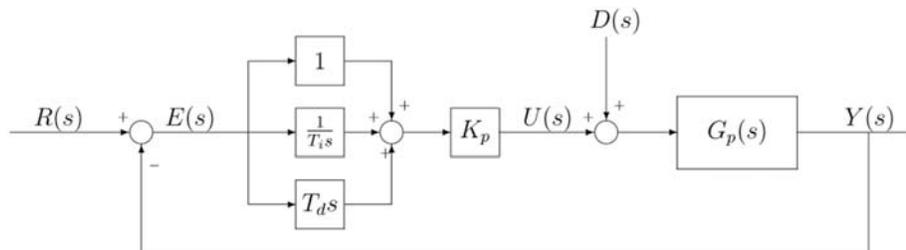


Fig. 2 – Esquema do Anel de Controlo PID

### 2.2.4 – Compensador de Avanço

Quando se pretende melhorar a resposta transitória, aumenta-se as Margens de Fase e de Ganho. O aumento da Margem de Fase é possível apenas com um ganho, no entanto, por vezes a precisão do anel de controlo diminui, piorando assim a resposta estacionária do sistema. Surge então o Compensador de Avanço.

O Compensador de Avanço é um controlador que tem como principal objectivo melhorar não só as características da resposta transitória mas também a estabilidade relativa, mantendo a precisão do anel fechado. Esses objectivos são conseguidos, como foi referido, aumentando as Margens de Estabilidade do sistema em termos de Resposta em Frequência ou no que respeita ao domínio do tempo, alterando a localização dos pólos dominantes de maneira a garantir certas especificações de desempenho como o sobreimpulso e o tempo de estabelecimento.

Desta forma, este compensador tem como expressão matemática:

$$G_c(s) = \frac{\alpha s + a}{s + a} \quad (12)$$

Assim, este controlador coloca um zero e um pólo no SPE, com o pólo mais afastado do eixo imaginário que o zero.

Este compensador é um caso particular do controlador PD, apresentado anteriormente, a baixas frequências, porque ao colocar o zero a frequência mais baixa que o pólo o LGR desvia-se para a esquerda alterando assim a dinâmica do sistema.

### **2.2.5 – Compensador de Atraso**

Ao contrário do controlador anterior, o Compensador de Atraso tem como objectivo melhorar a resposta estacionária, ou seja, melhorar a precisão do anel fechado, mantendo as características da resposta transitória. Este objectivo é conseguido mantendo as margens de estabilidade e mantendo aproximadamente a localização dos pólos dominantes do sistema. Desta maneira, os parâmetros deste controlador são determinados com base nos requisitos de Erros Estacionários.

A expressão do controlador é:

$$G_c(s) = \frac{s + a}{s + a/\alpha} \quad (13)$$

O que na prática significa colocar também um zero e um pólo no SPE, no entanto neste controlador é o zero que está mais afastado da origem.

## **2.3 – Métodos para o projecto de controladores**

### **2.3.1 – Via LGR**

A maneira mais simples de fazer o projecto dos controladores é recorrendo ao Matlab e à ferramenta *Sisotool*.

O método via LGR é especialmente utilizado quando se pretende garantir certas especificações de desempenho para a resposta no tempo do anel fechado. No entanto, no caso deste projecto, não se tem essas especificações directamente, tendo-se assim de recorrer unicamente a uma boa localização dos pólos.

É fundamentalmente a localização dos pólos da FT do anel fechado que determina a estabilidade e as características da resposta transitória do sistema. Desta forma é essencial que estes não se encontrem no Semi-Plano Direito, para garantir a estabilidade, e que tenham uma certa localização no plano complexo de maneira a dar as características pretendidas ao sistema, características essas que foram referidas no ponto 2.1.3.

### 2.3.2 – Ziegler-Nichols

Apenas uma breve referência a outros métodos que existem para o projecto de controladores PID, que, no entanto, não se revelam apropriados quando se trata de um sistema instável em anel aberto. Existem, então, dois métodos de Ziegler-Nichols que têm por semelhança não ser necessário saber a Função de Transferência do sistema:

#### ♦ Método do ganho crítico

Este método é caracterizado por levar o sistema ao limite de estabilidade. Consiste assim em considerar inicialmente apenas o ganho proporcional do controlador e determinar o valor deste no limite de estabilidade, determinando também o período crítico. Existem depois relações empíricas que permitem calcular os valores dos parâmetros dos controladores do tipo PID.

Como há processos que nunca chegam a instabilizar em anel fechado, este método nem sempre é aplicável.

#### ♦ Método da curva de reacção

Ao contrário do método anterior, este trata-se de um projecto em anel aberto. Desta maneira este método consiste na determinação de três parâmetros ( $\theta$ ,  $T$  e  $K$ ) através da observação da resposta a um degrau e fazendo a aproximação dessa resposta a um sistema de 1ª ordem com atraso no tempo. Os parâmetros dos controladores PID são determinados também através de relações empíricas propostas por Ziegler-Nichols.

Este método, para além de não ser aplicável a sistemas instáveis, como já foi referido, também não pode ser utilizado quando o sistema possui pólos complexos conjugados dominantes.

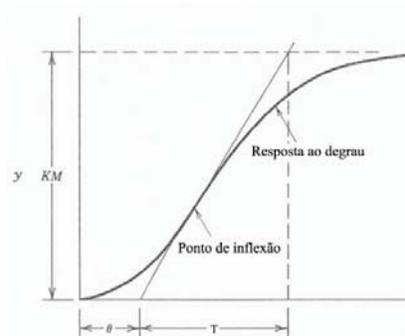


Fig.3 – Identificação dos parâmetros para o projecto via método da curva de reacção

### 3 - Rasteirinho

O Rasteirinho é um robot constituído basicamente por uma base metálica onde estão montados, na parte de baixo, dois motores e correspondentes baterias, ligados a duas rodas motrizes. Para estar devidamente apoiado está também montada uma “roda louca”. Na parte de cima da base encontram-se as duas placas de comando e dois apoios onde se deverá colocar o computador.

O funcionamento do Rasteirinho, do ponto de vista do utilizador, é bastante simples. O computador está ligado por USB a uma placa de conversão analógico/digital, digital/analógico (ADDA) da National Instruments. Esta, por seu lado, envia os sinais, já convertidos para analógicos, para uma placa de circuitos (PCB) que controla a velocidade dos motores, consoante a tensão enviada pela ADDA. Estando também ligado uma *Webcam* que filma o percurso e envia a imagem para o software de controlo.



Fig. 4 – Imagem do Rasteirinho

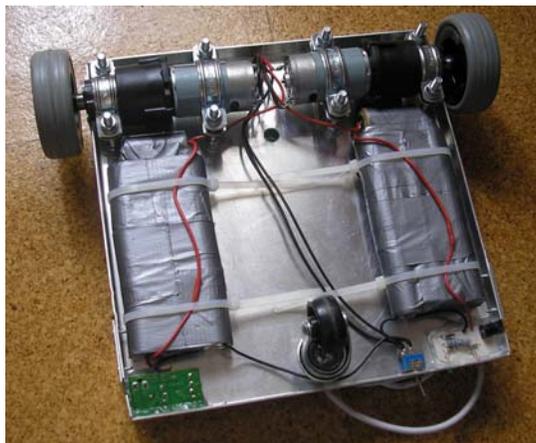


Fig. 5- Rasteirinho (visto de baixo)

## 4 - Passos para a execução do projecto

### 4.1 – Passos Iniciais

A primeira coisa a fazer em qualquer projecto é conhecer o objecto de trabalho, o que neste caso, significa verificar como o Rasteirinho reage quando não está sujeito a nenhuma solicitação específica. Por exemplo, é importante verificar se, quando a tensão aplicada aos motores é a mesma em ambos, o robot segue a direita ou se originalmente um dos motores dá uma tensão superior ao outro.

Após a afinação das tensões dos motores, colocou-se o Rasteirinho a seguir a linha preta contínua, utilizando apenas um controlador Proporcional, alterando assim o ganho a inserir no sistema até o robot atingir uma boa resposta. Entende-se aqui que uma boa resposta, só com um Proporcional, não é fazê-lo seguir obrigatoriamente toda a linha contínua, mas sim olhando para o gráfico retirado a partir da resposta ao degrau e ser possível retirar deste uma Função de Transferência.

### 4.2 – Função de Transferência para o Controlo de Posição

Através do gráfico obtido experimentalmente da resposta do sistema quando sujeito a um degrau, é possível retirar a localização dos pólos do anel fechado aproximando o gráfico obtido a um outro em que sejam conhecidos os seus parâmetros. A maneira mais fácil de o fazer é utilizando o Matlab existindo várias alternativas neste software para se fazer a referida aproximação.

Um método de aproximar os gráficos das funções é utilizando a função *lsim* do Matlab. O uso desta função exige que se dê inicialmente uma possível localização dos pólos e/ou zeros da FT em anel fechado e que se vá alterando as suas localizações de modo a conseguir aproximar o gráfico obtido ao pretendido. No entanto, para se saber o que se deve alterar, convém ter uma noção das consequências das diferentes localizações dos pólos/zeros da FT em anel fechado. Uma das hipóteses é olhando para o LGR, também disponível no Matlab através da função *rootlocus* ou da ferramenta *Sisotool*, e simultaneamente para a resposta no tempo e verificar o que acontece quando se mudam as localizações dos pólos (*Ver secção 2.1.3*).

Assim, tendo conseguido obter um pequeno troço do gráfico onde fosse possível compará-lo com um outro, independentemente de ser um sistema estável ou instável, passou-se à fase de determinação da FT através do processo *try-and-fail* (tentativa-erro). Através da função *lsim* do Matlab, variando tanto o ganho como a localização dos pólos da FT em anel fechado, assumindo que se tratava de um sistema de 2ª ordem, tentou-se então aproximar um gráfico ao obtido experimentalmente, tendo a noção que seria muito complicado obter gráficos exactamente iguais.

Determinados os pólos, para a obtenção da FT, há que fazer o trabalho inverso ao apresentado nas Bases Teóricas, isto é, tendo a FT anel fechado, pretende-se agora determinar a FT em anel aberto. Assim, utilizando a equação (5), consegue-se retirar por simples manipulação matemática a função de transferência do Rasteirinho,  $G_p$ :

$$G_p(s) = -\frac{G_{AF}(s)}{K(G_{AF}(s) - 1)} \quad (14)$$

Após se verificar no LGR que os pólos estão no sítio correcto pode-se passar à fase do projecto do controlador.

### **4.3 – Projecto do controlador**

Tendo-se a Função de Transferência do anel aberto do sistema em estudo, colocou-se esta na ferramenta *Sisotool* do Matlab, obtendo-se assim o LGR do sistema. Em seguida, olhando para o LGR e para a Resposta do sistema ao Degrau Unitário, experimentou-se os vários Controladores, adicionando pólos e/ou zeros de acordo com o controlador em estudo e alterou-se a localização dos mesmos até se encontrar a melhor resposta possível. Durante esta fase, deve-se ter particular atenção à localização dos pólos do sistema em anel fechado, pois estes devem estar sobre o Eixo Real, e ao valor do ganho que não deve ser elevado.

Encontrado o Controlador que aparenta ser mais eficaz retirou-se os respectivos parâmetros e implementou-se no Rasteirinho verificando o seu comportamento. Pode-se posteriormente fazer uma sintonia fina, para a obtenção de melhores resultados.

Também é importante ter a noção que o que pode parecer uma Resposta não ideal observando os resultados no Matlab, pode ser suficiente para o controlo do Rasteirinho.

### **4.4 – Controlo de Velocidade**

Tendo a trajetória do Rasteirinho controlada, passou-se à fase do controlo de velocidade. Nesta fase, não se conseguindo implementar um algoritmo de contagem dos traços, optou-se por recorrer ao Controlo Híbrido. De referir que por sistemas híbridos entende-se que sejam sistemas que são constituídos por parte contínua e por parte discreta. Sabendo isto, o controlo de velocidade foi feito da seguinte maneira:

Para alterar a velocidade precisava-se de alterar a tensão aplicada aos motores, alterando assim a dinâmica do sistema. No entanto, como as alterações nas tensões não eram substanciais, assumiu-se que a FT determinada anteriormente também não deveria ser muito diferente e desta maneira os controladores também deveriam ser próximos do obtido para o controlo de posição.

Sabendo que o Rasteirinho teria que passar por 4 troços diferentes em  $x$  tempo, apenas por *try-and-fail*, determinou-se os melhores controladores para compensar as diferenças nas tensões dos motores e conseqüente diferente resposta do sistema. Obtiveram-se assim quatro controladores diferentes, um para cada troço, que deveriam ser “programados” para entrarem em acção quando  $x$  tempo passasse. Desta maneira, o sistema sendo contínuo no tempo possui também “saltos”, aquando da mudança dos controladores, daí se chamar a este tipo de controlo, Controlo Híbrido. Isto é, uma

alteração da variável “tempo” provoca uma alteração brusca na variável de controlo do sistema (neste caso a tensão aplicada aos motores e os controladores correspondentes).

Os tempos estabelecidos para cada troço foram:

- Marca 1 (no final do 1º segmento em curva): 10 segundos
- Marca 2 (imediatamente antes da passadeira): 8 segundos
- Marca 3 (no final da recta): 6 segundos
- Marca 4 (final do circuito): 12 segundos

Para uma melhor compreensão do percurso, apresenta-se de seguida imagens do mesmo:



Fig. 6 – Imagem geral do percurso



Fig. 7 – Pormenor da última curva do percurso



Fig. 8 – Pormenor da zona da passadeira (vista no sentido oposto)

## 5 - Resultados obtidos

Inicialmente verificou-se que o Rasteirinho não precisava de tensões diferentes nos seus motores, pois as diferenças eram mínimas quando este seguia em linha recta. Assim, passou-se à fase de obtenção do gráfico de onde se retirou a FT. Este gráfico foi conseguido após várias tentativas e para uma tensão aplicada a ambos os motores de 1.1V e um ganho proporcional de 0.006.

Passando à utilização do Matlab e da função *lsim*, reduzindo o gráfico experimental apenas à parte que permite a comparação com um outro de parâmetros conhecidos, e após aproximação dos mesmos obteve-se o seguinte, estando indicado a azul a resposta experimental e a vermelho a conhecida.

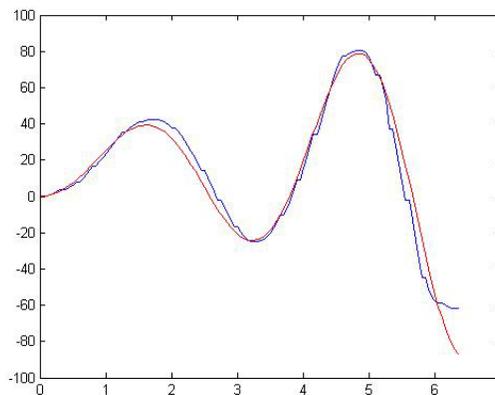


Fig. 9 – Aproximação da Resposta do Sistema

Conseguida uma boa aproximação determinou-se a FT correspondente ao gráfico apresentado através dos seus pólos.

Da utilização da função *lsim* do Matlab:

```
y=lsim(Num, Den, u, t)
```

Sendo

```
Den=poly([0.3+1.95j;0.3-1.95j]);  
Num=15*Den(3);
```

Resulta a FT em anel fechado:

$$G_{AF}(s) = \frac{58.39}{s^2 - 0.6s + 3.892} \quad (15)$$

E de seguida determinou-se a FT do anel aberto:

$$G_{AA}(s) = \frac{58.39}{0.006s^2 - 0.0036s + 0.327} \quad (16)$$

Com pólos em  $s = 7.6882$  e em  $s = -7.0882$

Olhando ao LGR, é possível verificar a localização dos pólos do sistema em anel fechado no SPD.

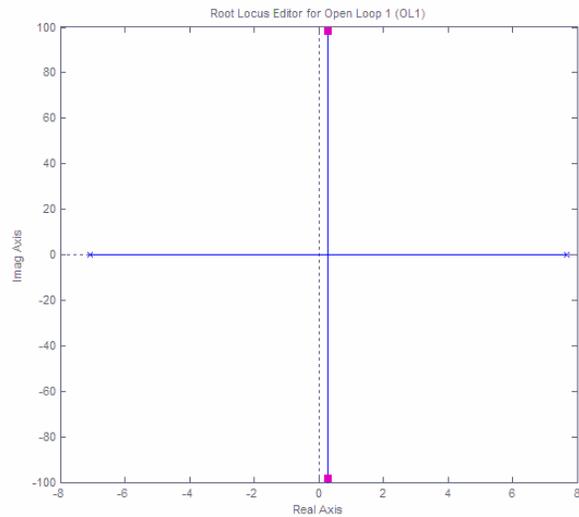


Fig. 10 – FT anel aberto antes do controlador

E na resposta ao degrau confirma-se que o sistema é instável:

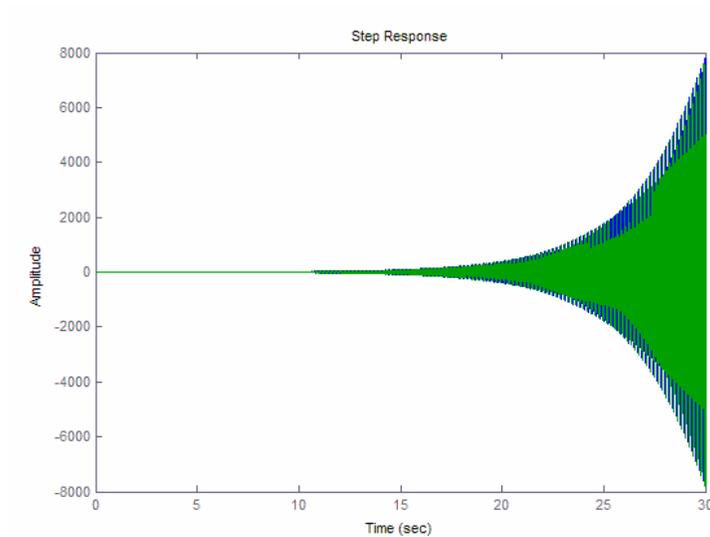


Fig. 11 – Resposta ao degrau antes do controlador

Para o Controlo de Posição, optou-se pelo Controlador PD, visto ser este o que originava uma melhor resposta. Assim, experimentou-se variar a posição do zero, de maneira a, olhando sempre para o LGR e para a resposta ao degrau em simultâneo, obter a melhor resposta possível.

Obtiveram-se assim os seguintes resultados:

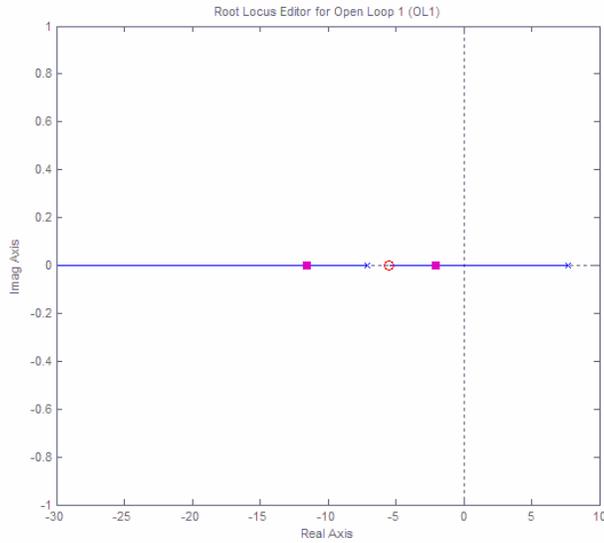


Fig. 12 – LGR com compensador para a posição

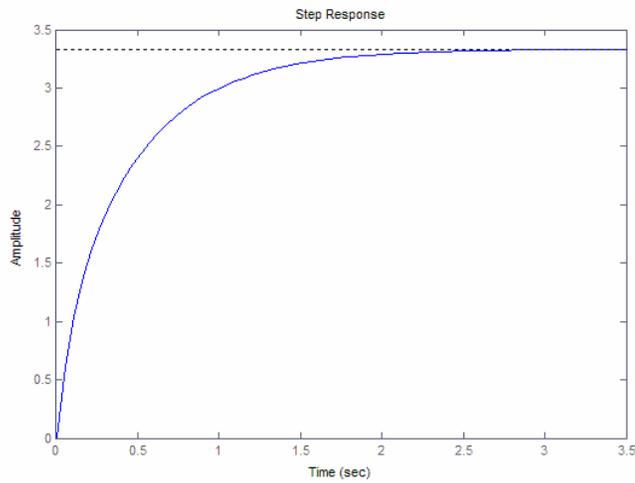


Fig. 13 – Resposta ao degrau com compensador PD

Considerada esta como uma possível boa resposta, retirou-se directamente a sua FT, tendo-se obtido:

$$G_{cPD}(s) = 0.008 \cdot (1 + 0.18s) \quad (17)$$

Para o controlo de velocidade, considerando que a FT não seria alterada em muito com a alteração da tensão  $V_n$  aplicada aos motores, este controlo foi feito com base no processo *try-and-fail*, não tendo sido retirados quaisquer dados do LGR, visto que as diferenças sendo mínimas só seriam mesmo perceptíveis se seriam bons controladores experimentando directamente no Rasteirinho e comparando a sua resposta com os tempos efectuados nos diferentes troços.

Os diferentes controladores utilizados para cada troço foram os seguintes:

1º Troço:

$$V_n = 1.28V$$

$$G_{c1}(s) = 0.008 \cdot (1 + 0.2s) \quad (18)$$

2º Troço:

$$V_n = 1.38V$$

$$G_{c2}(s) = 0.005 \cdot (1 + 0.65s) \quad (19)$$

3º Troço:

$$V_n = 1.5V$$

$$G_{c3}(s) = 0.006 \cdot (1 + s) \quad (20)$$

4º Troço:

$$V_n = 1.1V$$

$$G_{c4}(s) = 0.005 \cdot (1 + 0.2s) \quad (21)$$

## 6 - Comparação de resultados entre Controladores

Para o Controlo de Posição, utilizando a ferramenta *Sisotool*, experimentou-se os vários tipos de controladores. Apresenta-se de seguida os melhores resultados destas experiências para cada um dos controladores, sendo fácil de verificar que alguns deles nunca estabilizariam o sistema.

### 6.1 – PID

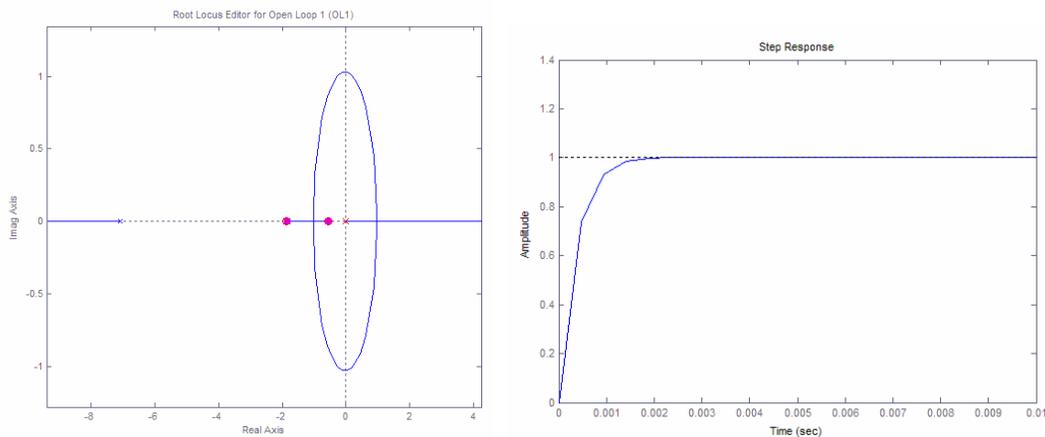


Fig. 14 – LGR e Resposta ao degrau unitário do sistema controlado com PID

$$G_{PID}(s) = 0.3 \cdot \frac{(1 + 1.8s)(1 + 0.53s)}{s} \quad (22)$$

De onde resulta, por simples manipulação matemática:

$$K_p = 0.699$$

$$T_d = 0.41$$

$$T_i = 2.33$$

Este controlador, a funcionar, originaria uma resposta muito rápida do sistema e sem Erro Estacionário. Ao diminuir o ganho  $K_p$ , apareceria na resposta um sobreimpulso indesejável.

## 6.2 – PI

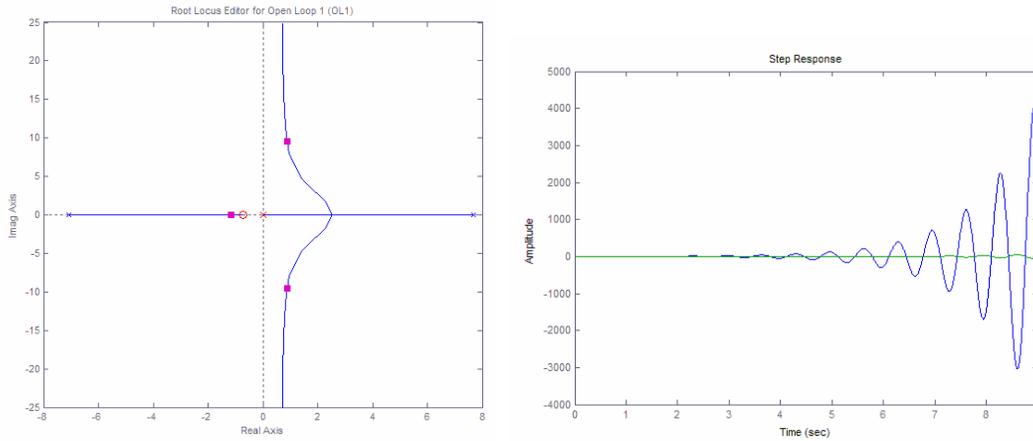


Fig. 15 – LGR e Resposta ao degrau unitário do sistema controlado com PI

$$G_{PI}(s) = 0.01 \cdot \frac{(1 + 1.4s)}{s} \quad (23)$$

Com os parâmetros:

$$K_p = 0.007$$

$$T_i = 1.4$$

Por mais que se aproxime o zero do eixo imaginário é impossível colocar os pólos no SPE, logo não é possível controlar o sistema com este controlador.

## 6.3 – PD

Para o controlador PD, haviam duas hipóteses consoante a localização do zero. Uma delas já foi apresentada como tendo sido considerada a melhor resposta. Apresenta-se agora a outra solução, com o zero do controlador à esquerda do pólo da FT.

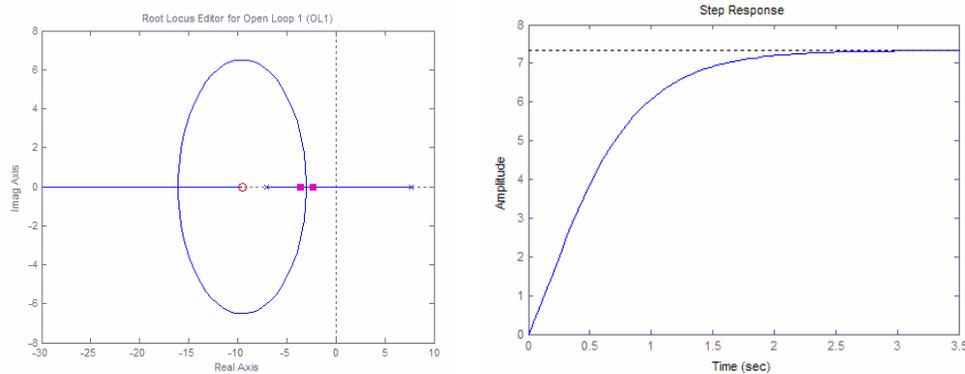


Fig. 16 – LGR e Resposta ao degrau unitário do sistema controlado com PD

$$G_{PD}(s) = 0.006 \cdot (1 + 0.1s) \quad (24)$$

Com os parâmetros retirados directamente:

$$K_p = 0.006$$

$$T_i = 0.1$$

Esta resposta, para valores de ganho baixos, verifica-se que apresenta um Erro Estacionário elevado. Para valores mais elevados, apresentaria um pequeno sobreimpulso.

### **6.4 – Compensador de Avanço**

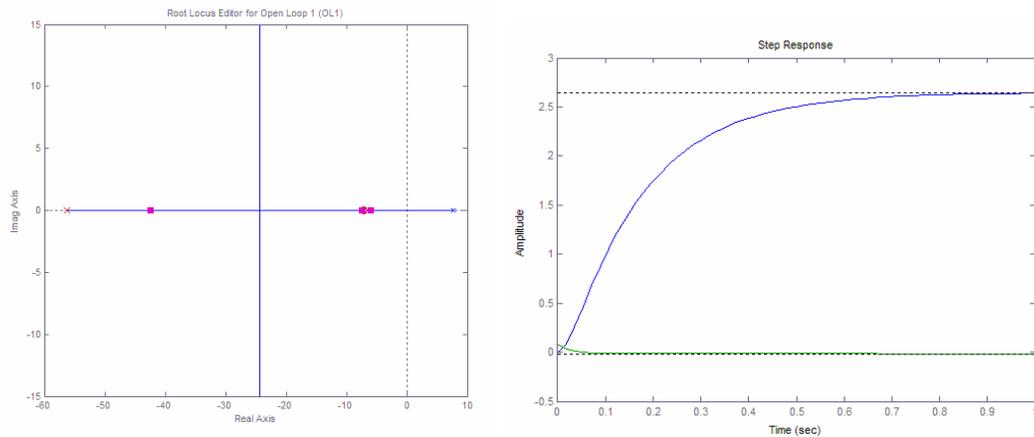


Fig. 17 – LGR e Resposta ao degrau unitário do sistema controlado com Compensador de Avanço

$$G_{C_{Avanço}}(s) = 0.009 \cdot \frac{(1 + 0.14s)}{(1 + 0.018s)} \quad (25)$$

Este compensador consegue estabilizar o sistema, apresentando no entanto um Erro Estacionário só controlável com o aumento da constante  $K_p$ .

## 6.5 – Compensador de Atraso

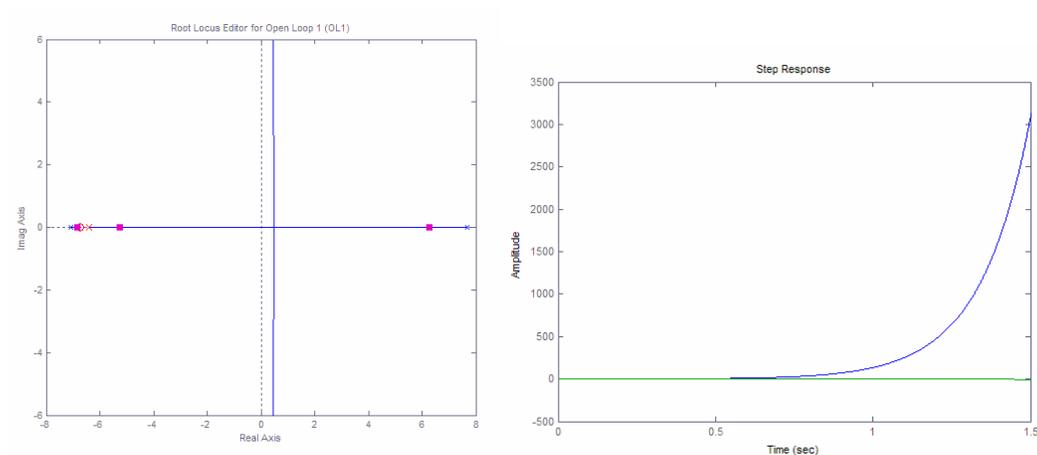


Fig. 18 – LGR e Resposta ao degrau unitário do sistema controlado com Compensador de Atraso

$$G_{Atraso}(s) = 0.002 \cdot \frac{(1 + 0.15s)}{(1 + 0.16s)} \quad (26)$$

Verifica-se que é impossível controlar o sistema com um Compensador de Atraso, visto que, por mais que se aproxime o zero do pólo sem nunca inverter as suas posições, os pólos do anel fechado vão estar sempre no SPD.

## 7 - Resultados obtidos na Competição

No que respeita à execução prática deste trabalho, isto é, à competição em si, os tempos obtidos em cada troço, com os controladores e tensões aplicadas referenciados no ponto 5, foram os seguintes:

	Tempo Obtido [segundos]	Tempo Estabelecido [segundos]
Marca 1	11.1	10
Marca 2	19.1	18
Marca 3	24.7	24
Marca 4	32.9	36

Tab. 1– Comparação entre os tempos obtidos e os estabelecidos

No final, houve uma penalização de 5 segundos por se ter coberto a passadeira.

Das tentativas efectuadas no decorrer da competição só numa delas o Rasteirinho terminou a prova com sucesso. Assim, retiraram-se os dados relativos ao Erro em cada instante dessa tentativa (que posteriormente foram utilizados pelos juizes da prova para calcular o Erro Quadrático Médio de seguimento) e, tendo em conta o Período de Amostragem de 0.05s, obteve-se o gráfico seguinte:

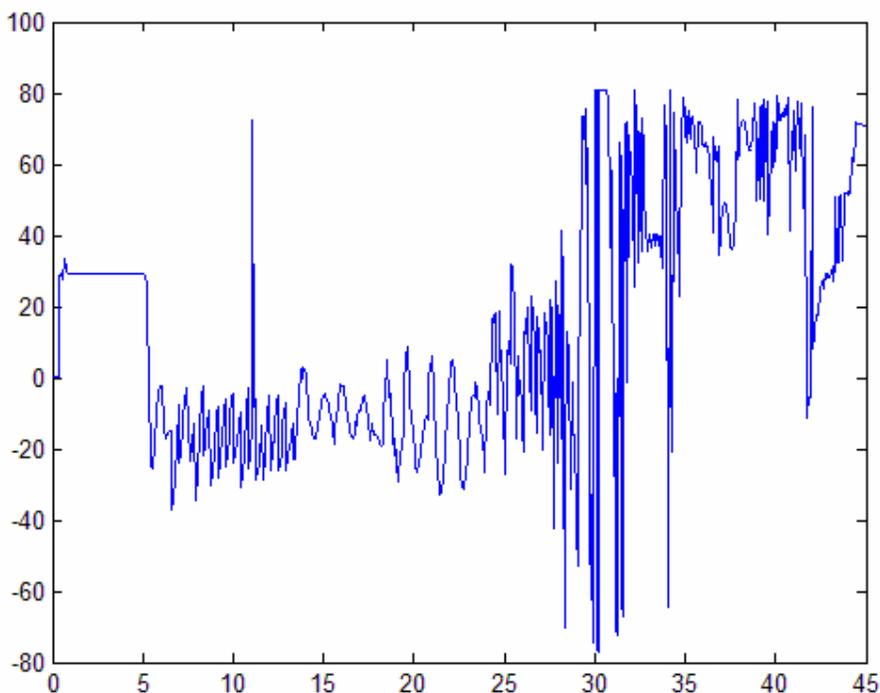


Fig. 19 – Gráfico Tempo.vs. Pixel

Esta figura representa o Erro, em pixéis, e nela podem-se identificar as diferentes zonas do percurso. De referir que a escala de tempo apresentada no gráfico não corresponde ao tempo real, mas sim ao tempo do Matlab, daí não se poder fazer uma comparação entre os tempos efectuados (apresentados na Tab.1) e o gráfico obtido.

Antes da análise por troços, este gráfico dá a noção de qual o desvio “natural” do Rasteirinho. Isto é, em secções o Erro aparece na zona negativa do eixo das coordenadas, noutras aparece como sendo positivo. Assim, verifica-se facilmente que raramente o Rasteirinho segue a linha centrado com esta, evidenciando-se desta maneira o Erro Estacionário, pois em todos os troços há oscilações, mais ou menos evidenciadas, que deveriam ser em torno de zero, que é o meio da imagem.

A primeira parte do gráfico, muito bem definida entre 0 e 5 unidades de tempo do Matlab (u.t.Matlab), corresponde ao tempo de espera para que o Rasteirinho visse a linha em condições de evitar o *Derivative Kick*, e como tal o Erro mantém-se constante. Apenas de notar que este Erro que aparece não é zero pelo facto do Rasteirinho não estar totalmente centrado com a faixa preta.

Iniciando o seu percurso, a zona entre 5 e 14 u.t.Matlab corresponde à primeira curva. Nesta parte salta à vista um pico, claramente devido a pequenas manchas que foram captadas pelo Rasteirinho e que não faziam parte da trajectória.

Entre as 14 e as 19 u.t.Matlab está-se na zona da primeira parte da recta e como era de esperar a variação do Erro não é tão pronunciada.

Na segunda parte da recta, a zona que contém a passadeira, entre os 19 e os 24 u.t.Matlab, verifica-se que o Erro manteve-se relativamente próximo do verificado na primeira parte da recta, apenas com maior “amplitude” possivelmente devido à mudança do Controlador, e este ter uma Constante Derivativa superior. Verifica-se também que nesta zona, como era de esperar, o efeito da passadeira não se fez sentir na variação do Erro, pelo facto desta estar coberta, sendo só visualizado o preto da linha a tracejado e não o restante preto característico dessa zona.

Dos 24 a aproximadamente 40 u.t.Matlab verifica-se o troço mais complicado, a curva final do trajecto. Pode-se verificar esse facto, tanto pela verificação dos tempos efectuados (foi a zona onde os tempos efectuado e pretendido estão menos próximos) como pela análise do gráfico do Erro. Pela análise deste último, primeiro constata-se que a mudança do Controlador se deu antes da chegada à curva, pois deu-se aos 24 e a entrada na curva aparenta ter sido por volta das 30 u.t.Matlab, altura em que se verifica uma grande oscilação na zona inicial deste troço. Já na parte final é notório que o Rasteirinho não fez a curva centrado com linha, mas sim, desviado para a esquerda, como que fazendo a curva “por dentro”, quase no limite da sua captura da imagem.

A zona dos 40 até ao final não corresponde a uma zona em estudo pois a prova já se encontrava concluída, sendo esse tempo, o tempo que se demorou até desligar o robot.

## 8 - Discussão de resultados

Através da análise dos resultados obtidos na realização deste trabalho, a primeira observação que se pode fazer é que só com um Proporcional o sistema é instável, visível através do LGR ou da Resposta ao Degrau, mas também no comportamento do Rasteirinho ao oscilar bastante, até perder de vista a referência (linha preta contínua).

No que toca à aproximação dos gráficos, aqui reside grande parte da falta de precisão deste método de controlo. Isto porque, até parecendo ser uma boa aproximação, o número de picos é pequeno (apenas dois) e ambos os gráficos não estão exactamente sobrepostos. No entanto, seria difícil fazer melhor só com dois pólos complexos conjugados. Podia-se ter juntado um zero e tentado melhorar, mas com tantos outros atritos existentes, a ser referidos mais adiante, esta foi considerada uma boa aproximação.

Quanto ao projecto dos Controladores, dos cinco inicialmente propostos escolheu-se o PD como sendo o que melhor se aplicava a este projecto. Isto porque a componente Integral, que eliminaria o Erro Estacionário, provocava um comportamento “estranho” do Sistema, pois este fazia mudanças bruscas de direcção e não apenas pequenas compensações. Analise-se então de seguida o porquê da escolha do Controlador Proporcional-Derivativo.

Uma vez que existiria o problema do Erro Estacionário, a primeira ideia era que seria necessário um controlador PI e que para “chamar” os pólos para o SPE, seria necessário um PD. Logo, a primeira hipótese recaiu sobre o Controlador PID. Daí ser um facto importante verificar o porquê de, na execução deste projecto, o PID não ter funcionado. Este problema não podia ter sido identificado apenas pela observação do LGR e da Resposta ao Degrau mas somente quando se implementou o PID no Rasteirinho. Uma possível razão para o comportamento do Rasteirinho, quando controlado com o PID, ter sido o verificado é o efeito *reset-windup*. Isto porque, no projecto deste controlador através da ferramenta *Sisotool*, para se conseguir obter os pólos no eixo real e se conseguir uma resposta de certa forma rápida era necessário um valor de ganho bastante superior aos utilizados no resto do projecto. Esse valor de ganho pode ter levado a uma saturação da acção de controlo. Ora, quando ocorre essa saturação e há uma acção Integral no controlador, o sistema fica mais lento a responder e com maior sobreimpulso, resultando daí a “estranha” resposta do Rasteirinho. Há ainda outra possibilidade que é a existência de um pequeno sobreimpulso, impossível de evitar no projecto do PID (não muito perceptível na Fig.14), e este efeito, reunido com o facto de a resposta ser extremamente rápida, talvez pudesse originar o referido comportamento. No entanto, o que leva a pensar que terá sido a primeira hipótese é que após o sucedido, analisou-se o comportamento do Rasteirinho, ligando-o e segurando-o também e verificou-se que um dos motores estava parado sendo esse um sinal que se tinha atingido a saturação.

Estando determinado o porquê de não se ter utilizado o controlador PID, falta verificar o porquê da escolha do PD em detrimento dos Compensadores.

É fácil verificar através do LGR que o Compensador de Atraso, assim como o PI, não consegue estabilizar o sistema, sobrando assim apenas o Controlador de Avanço. Sendo este um caso particular do PD, as diferenças são mínimas, não tendo grande diferença nos efeitos provocados no Rasteirinho. Daí a escolha efectuada.

Quando se analisa o LGR e a Resposta ao Degrau depois de se controlar o sistema com o Controlador PD, verifica-se que, como era de esperar, o Erro Estacionário é elevado, no entanto, isso não foi verificado no Rasteirinho, pois ele aparentou estar relativamente centrado com a linha preta. Isto deverá estar relacionado com os atritos existentes e os “erros” cometidos ao longo do projecto. Ainda na Resposta ao Degrau Unitário, retira-se um tempo de estabelecimento superior ao dos outros controladores, o que também pode parecer elevado, mas novamente isso não foi verificado.

No que respeita ao Controlo de Velocidade, como já foi referido, utilizaram-se quatro controladores diferentes, tendo por base o controlador obtido para o Controlo de Posição. A alteração dos parâmetros  $K_p$  e  $T_d$  foi feita apenas com base em experiências no Rasteirinho e com o traçado da pista. Assim, optou-se por aumentar o  $T_d$  na parte das rectas, visto nessas zonas não serem necessárias grandes “compensações” de trajectória e para dessa forma se tentar evitar que o Rasteirinho “fugisse” quando visse alguma zona preta fora da trajectória. Já os valores de  $K_p$  foram decididos tendo em conta a alteração das tensões aplicadas aos motores e se se pretendia que o Rasteirinho fosse mais ou menos rápido em cada troço. Como o valor da constante proporcional influencia a tensão aplicada aos motores, tentou-se arranjar um compromisso entre os dois que resultasse nos tempos pretendidos para cada troço.

De referir também que o último troço foi o que gerou maiores dificuldades para conciliar os tempos pretendido e efectuado, por dois motivos principais. Primeiro o facto de não se ter conseguido ultrapassar a zona da passadeira, no início do terceiro troço, fez com que o número de experiências na última fase não fosse o suficiente. A outra razão para não se ter conseguido atingir esse objectivo é que, apesar dessa zona ser semelhante à primeira, o facto de o Rasteirinho já ir em andamento quando inicia a entrada na curva e o trajecto seja no sentido oposto fez com que a imagem vista na câmara fosse diferente. Isto é, a imagem visível para o Rasteirinho nunca foi devidamente “centrada” de maneira a que o robot ia sempre desviado para a esquerda. Como tal foi mais difícil conciliar os valores do ganho, constante derivativa e tensão aplicada neste último troço e daí os piores resultados nesta zona quando vistos os tempos.

Para ultrapassar a zona da passadeira tentou-se várias soluções. Em primeiro lugar, experimentou-se aumentar a Constante Derivativa do controlador, tentando que a resposta fosse mais lenta nessa zona e o Rasteirinho quando fosse a reagir aos traços da passadeira, esta já estivesse ultrapassada. No entanto, esta hipótese não resultou pois primeiro não se podia aumentar  $T_d$  demasiado, pois poderia instabilizar o sistema mantendo o valor do ganho, depois porque bastava que o Rasteirinho fosse ligeiramente a oscilar para que quando passasse nessa zona ele se perdesse.

Também se experimentou aplicar um saturador nas saídas das tensões para tentar evitar que o Rasteirinho pudesse “guinar” quando via uma referência muito deslocada

para um dos lados. Este método mostrou-se ineficiente visto que, mesmo aplicando o saturador só nas secções rectas, o Rasteirinho perdia capacidade de se controlar.

Em seguida, tentou-se fazer algo semelhante mas desta vez o *software* ia guardando em memória o último valor calculado para a tensão e ignorava qualquer ordem que se distanciasse muito do último obtido. Este método provou-se ineficaz por tornar o computador extremamente lento, impedindo o controlo do Rasteirinho.

Por último decidiu-se criar uma rotina em Matlab que lia a matriz binária da imagem a preto e branco e punha a branco todas as linhas com mais do que um determinado número de pixéis pretos. O objectivo era impedir o Rasteirinho de ver a passadeira e continuar a trajectória que trazia anteriormente, mas mais uma vez, devido ao peso do Matlab, esta rotina esgotava os recursos do computador.

Assim, todas as tentativas foram infrutíferas, optando-se por cobrir essa zona de maneira a cumprir todo o percurso.

Para finalizar, há que ter em conta que a fase de projecto tem como nuance o facto de aquela não ser a verdadeira FT do Sistema mas sim uma aproximada. Isto porquê? Porque há muitos factores (e alguns deles impossíveis de controlar) que influenciam o comportamento do Rasteirinho: O estado das baterias, a posição da câmara, o ruído (visível no gráfico obtido experimentalmente), as oscilações do piso... Desta maneira, todo o projecto dos controladores para o controlo de posição via Matlab foram aproximadas e em consequência também os controladores para a velocidade o foram.

## 9 - Pontos a ter em conta para a execução do Projecto

Durante a execução deste projecto, foram várias as situações que tiveram de ser tomadas em conta para que fossem atingidos os objectivos. Apresentam-se aqui alguns dos principais problemas encontrados e suas possíveis resoluções.

### ▪ *Baterias*

Um dos principais problemas encontrados durante o projecto foi o desconhecimento do estado das baterias do Rasteirinho. Isto porque este é um factor decisivo na caracterização do sistema, ou seja, se as baterias não estiverem sempre mais ou menos nas mesmas condições, o comportamento do sistema é completamente alterado, sendo assim mais difícil de fazer o controlo.

Sabendo das dificuldades, tentou-se carregar totalmente as baterias, no entanto isso não foi possível já que, quando se regressou ao laboratório, as baterias do Rasteirinho encontravam-se desligadas da corrente, sendo assim impossível de saber ao certo quantas horas estas carregaram. Como se conseguiu retirar a Função de Transferência nesse dia, optou-se por fazer carregamentos de quatro a seis horas de maneira a que as baterias se mantivessem ao mesmo nível de quando se fez a identificação do sistema.

Este método foi eficaz, no entanto, o ideal será fazer o carregamento total das baterias para assim não se estar a trabalhar num total desconhecimento do estado de carga das mesmas.

### ▪ *Posição da Webcam*

A captura correcta da imagem provou-se também algo complicado.

Por um lado, a posição em que se colocava a *Webcam* influenciava a qualidade da imagem recolhida e também o número de traços vistos. Se a câmara estivesse demasiado baixa, esta podia apanhar a sombra do robot ou não apanhar traços suficientes para garantir um seguimento satisfatório. Levantar demasiado a câmara levava à captura de elementos indesejáveis na imagem, que podem “puxar” o Rasteirinho para longe da linha.

Este factor influenciou bastante quando se tentou retirar um gráfico passível de se identificar uma FT, pois o facto do Rasteirinho perder de vista a linha preta devido aos reflexos (mesmo não instabilizando) e fazer o reconhecimento de faixas cinzentas ou de outros “ruídos” presentes no piso como preto fazia com que o cálculo do centróide fosse alterado e o gráfico não representasse o comportamento real do Rasteirinho.

Para se tentar diminuir o efeito de ruído branco e fazer com que as faixas cinzentas presentes ao longo do percurso não fossem reconhecidas como pretas, procedeu-se à implementação de uma rotina de binarização da imagem onde se podiam ajustar os valores de captura para cada cor (vermelho, verde e azul). Não se conseguindo eliminar todos os ruídos, foi possível otimizar a captura às necessidades principais do projecto, não sendo, no entanto, suficiente para que se conseguisse fazer a contagem dos traços já

que um reflexo no meio dum traço fazia com que o software contasse dois traços em vez de um.

- ***Placa da National Instruments***

Um problema algo insólito que surgiu foi as paragens repentinas do Rasteirinho sem motivo aparente. Após análise chegou-se à conclusão que se tratava de um erro, provavelmente por mau contacto, da placa da National Instruments, que perdia a comunicação com o computador.

A solução a este problema é retirar o cabo USB do computador e voltar a ligá-lo.

- ***Perturbações externas***

Como principais perturbações no anel de controlo, surgem as imperfeições no piso do percurso que levavam a que o Rasteirinho tivesse por vezes uma captura inadequada da imagem já que, para além de alterarem o movimento do robot, faziam com que a tampa do computador abanasse, alterando a posição da câmara, podendo levar à instabilização.

Nesta situação, não há muito a fazer, se as perturbações forem ligeiras, o Controlador deve fazer com que o sistema seja capaz de as ultrapassar. Caso contrário, uma vez que não se conseguem medir essas perturbações, torna-se complicado controlar o Rasteirinho.

- ***Derivative Kick***

Durante todo o processo de colocar a funcionar o Rasteirinho teve que se ter cuidado com o problema do *Derivative Kick*. Este problema foi mais evidenciado uma vez que o controlador escolhido foi o PD. Assim, como a informação não entrava no anel de controlo ao mesmo tempo que o sistema era ligado, o Rasteirinho demorava algum tempo a ver a linha. Para evitar este problema, uma vez que não se implementou a estratégia de controlo P-D, decidiu-se ligar primeiro o software de controlo e só quando o Rasteirinho estivesse a ver a linha é que o resto do sistema entrava em funcionamento. Este procedimento fez com que o Rasteirinho só começasse a responder à imagem depois desta estar a ser correctamente recolhida, ou seja, quando o valor do Erro era praticamente nulo.

- ***Contagem do tempo, em Matlab***

Quando se procedeu ao controlo de velocidade, era necessário fazer uma contagem do tempo, para se estabelecer a altura em que se daria a troca dos controladores. Para isso, decidiu-se em primeira instância, utilizar como referência o tempo do Matlab. Aquando da utilização desta referência, chegou-se à conclusão que o tempo do Matlab

não correspondia à realidade e que este dependia da frequência de trabalho do software que não é constante.

Primeiramente, tentou-se fazer uma relação entre o tempo do Matlab e o tempo real, no entanto, como a frequência não é constante, isso não foi possível, sendo a única solução fazer este controle por tentativa-erro.

▪ **PID do Matlab**

A ferramenta *Simulink* do Matlab já contém na sua biblioteca um bloco PID (Fig. 20). No entanto, a forma como este bloco está montado não corresponde à implementação leccionada. Por esta razão criou-se um “*My PID Controller*” que correspondesse à implementação conhecida (Fig. 21), sendo essa a forma encontrada para se poder usar os parâmetros determinados através do *Sisotool*.

No caso do PD, não existem grandes diferenças, no entanto, como já se tinha implementado o “*My PID Controller*” optou-se por simplesmente eliminar a componente integral deste bloco, resultando no esquema da Fig. 22.

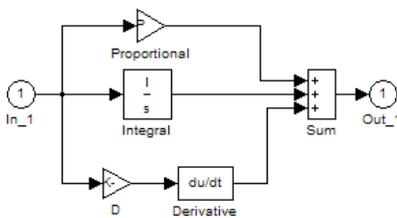


Fig. 20 – Esquema do PID disponível no Matlab

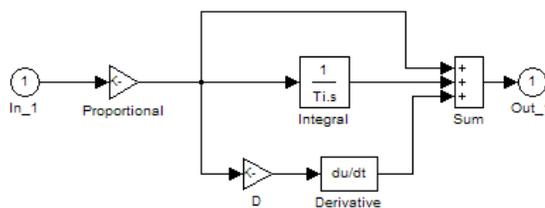


Fig. 21 – Esquema “*My PID Controller*”

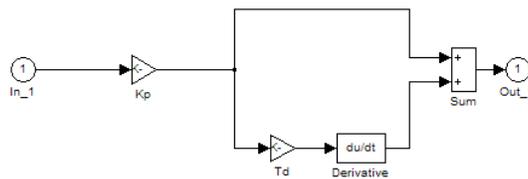


Fig. 22 – Esquema do PD utilizado

- ***Período de amostragem***

Um ponto importante é a definição do Período de Amostragem. Este tem de estar definido de igual modo em dois lados, para não haver conflitos no software. Assim, no bloco “Video Input”, onde se define a imagem retirada da *Webcam*, esta informação está definida em “Frame Rate”. Tendo definido os *frames-per-second* que a *Webcam* capta, é necessário passar este valor para *segundos* fazendo o inverso dos fps e posteriormente defini-lo na seguinte localização:

Menu *Simulation – Configuration Parameters – Fixed-step size*

Para exemplificar, neste projecto determinou-se um *Frame Rate* de 20fps o que resultou num Período de Amostragem de 0.05s.

## 10 - Conclusões

Após o término deste trabalho, chega-se à conclusão que o objectivo principal foi atingido. Isto é, conseguiu-se fazer o controlo do Rasteirinho de maneira a este seguir a linha a tracejado e com diferentes velocidades para cada troço. No entanto, o controlo de velocidade não foi efectuado através do algoritmo de contagem dos traços como estava inicialmente previsto principalmente devido a não se conseguir controlar os reflexos, que contariam como zonas brancas, e zonas do piso que eram reconhecidas como preto, optando-se assim por recorrer ao Controlo Híbrido.

Um dos problemas verificados no uso dos controladores foi o Controlador PID. Como foi referido na secção Discussão de Resultados, pensa-se que este não tenha resultado devido ao efeito *reset-windup*. Poderia se ter tentado evitar este efeito, através do “mecanismo” *reset anti-windup*, fazendo a realimentação não linear da saída do controlador, de maneira a que sempre que ocorresse a saturação a acção integral fosse “eliminada”. No entanto, sendo que esses conceitos não fazem parte de conhecimentos adquiridos nem na disciplina de Controlo de Sistemas nem noutra disciplina precedente, optou-se por não dar grande importância a essa situação, tentando-se assim outros controladores.

Quanto ao controlo de velocidade, neste trabalho o método utilizado pressupôs que se conhecesse bem o trajecto e se soubesse de antemão quais os tempos a cumprir de maneira a se poder fazer os testes correspondentes. Isto é, como a variável que determina a alteração brusca na variável de controlo do sistema é o tempo, se esta transição for feita cedo demais, ou seja, se a tensão aplicada não for suficiente para que o tempo fosse cumprido, como os controladores para cada troço são diferentes, o Rasteirinho poderia perder de vista a referência. Se se conseguisse adoptar uma estratégia de Controlo Robusto, isto é, se apenas um controlador fosse capaz de controlar todo o percurso, mesmo com as alterações das tensões aplicadas, esse efeito já não se deveria fazer sentir.

Em termos gerais, pode-se concluir neste trabalho que nem todos os problemas de controlo têm de ser resolvidos individualmente. Por vezes, o ideal é olhar directamente para o conjunto de problemas a resolver. Isto é, neste caso, o Erro Estacionário existia apesar de nem sempre muito evidente, e, no entanto, conseguiu-se controlar o Rasteirinho apenas com um PD, uma vez que o termo Integral estragava a dinâmica do sistema e assim conciliando o valor do ganho, para reduzir o Erro, com a localização dos pólos do anel fechado conseguiu-se uma boa resposta.

## 11 - Bibliografia

[1] Botto, M. A., *Controlo de Sistemas - 3ºano e 4º/5º anos do ramo de Automação e Robótica*, AEIST, Fevereiro 2007

[2] Helsinki University of Technology – Control Engineering Laboratory – Analog Control – <http://www.control.hut.fi/Kurssit/AS-74.2111/index.en.html>

[3] Clube de Automação e Robótica – <http://www.dem.ist.utl.pt/~carclub/Clube/>

## **12 - Agradecimentos**

Um agradecimento especial, ao Professor Miguel Ayala Botto, por todo o tempo despendido para nos ajudar, apoiar e esclarecer ao longo da realização deste trabalho.

Agradece-se também ao Professor José Borges, pela ajuda prestada.

A todos os colegas, participantes ou não da competição, que nos ajudaram na realização deste projecto, e que nos disponibilizaram o Software.