Universidade de Coimbra

Faculdade de Ciências e Tecnologia

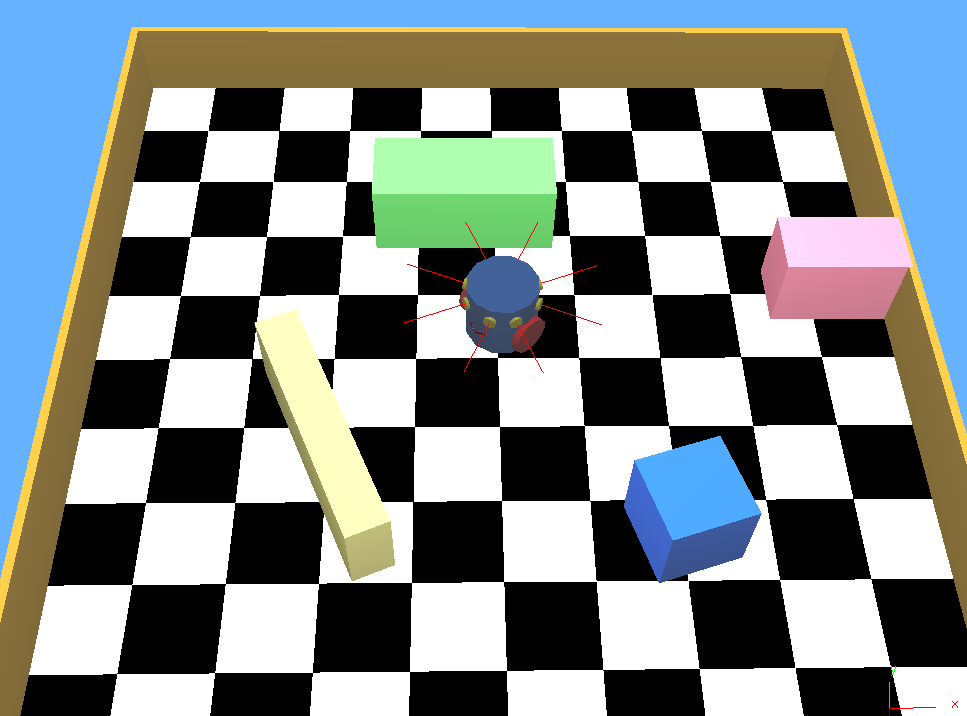
Licenciatura em Engenharia Informática

*Relatório Nº 2*

*Introdução à Inteligência Artificial, 1º Semestre*

**Veículos de Braitenberg**

**…Tune it & Test it…**



**Trabalho realizado por:**

* **Ivo Daniel Venhuizen Correia, nº2008110814**
* **João Pedro Gaioso Barbosa, nº2008111830**

Coimbra, 21 de Março de 2011

Índice

[Uma Breve Introdução 2](#_Toc288415679)

[A Nossa Implementação 2](#_Toc288415680)

[As funções de activação 2](#_Toc288415681)

[Cenários de Teste 2](#_Toc288415682)

[Contributos Individuais 3](#_Toc288415683)

[Perspectivas Futuras 3](#_Toc288415684)

[Bibliografia 4](#_Toc288415685)

# Uma Breve Introdução

Com os sensores e fontes devidamente planeados e testados, partimos para a segunda meta do trabalho. Foi-nos pedido a implementação de funções de activação que ultrapassassem os limites impostos pela opção exclusiva de um esquema linear, criando limites laterais, superiores e inferiores para as referidas funções.

Com estas novas funcionalidades, explorámos diversas capacidades dos veículos de Braitenberg, muitas das quais se tornaram úteis para uma melhor definição do produto final e perspectiva de dificuldades que iremos encontrar. Deste modo, uma vez mais o relatório segue a sua estrutura inicialmente estabelecida, com uma análise da implementação das funções de activação, cenários de teste, contributos individuais, para concluir com as nossas previsões para a meta final.

# A Nossa Implementação

## As funções de activação

De modo a estender o nosso leque de possibilidades, foi-nos sugerido a implementação de duas novas funções de activação para além da linear, nomeadamente uma função logarítmica e gaussiana. Para além destas, criámos também a função exponencial, sendo que no final ficámos com quatro escolhas diferentes para as funções de activação dos sensores.

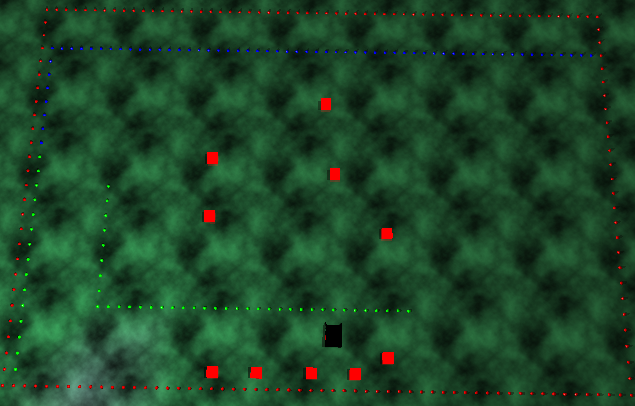
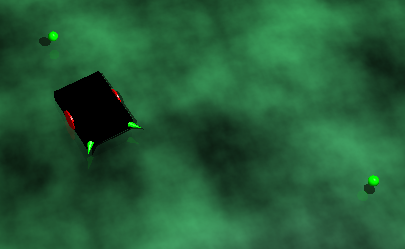
De modo a facilitar a manipulação do código, implementámos uma nova classe, chamada de *BraitenbergActivationObject*, que contém atributos relativos ao tipo de função de activação, os limites laterais de activação (*threshold*), assim como o máximo e mínimo admissível da resposta (limites superiores e inferiores). Uma vez definidos estes parâmetros, associamos um objecto desta classe a cada sensor. Esta classe contem ainda um método denominado *activate()* que é invocado, por sua vez, pelo método *iterate()* dos vários sensores. O que este método de activação simplesmente faz é receber um valor de força vindo do *iterate()*, aplicar uma dada função de activação (anteriormente definida aquando da criação do objecto), verificar os limites impostos pelo programador e devolver o valor de saída de acordo com a entrada disponibilizada.

## Cenários de Teste

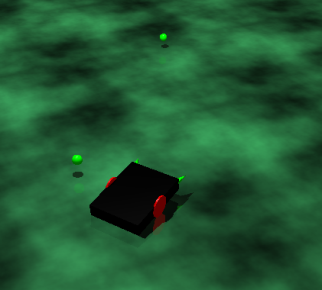
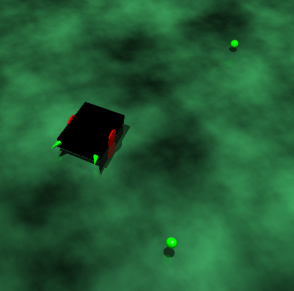
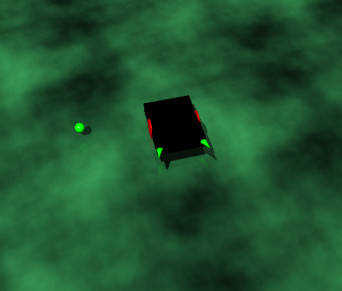
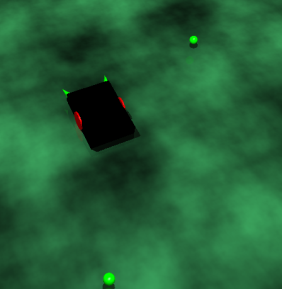
Como cenários de teste para esta meta, começámos pelo cenário onde era exigido que o veículo desenhasse um elipse em torno de dois focos (figura 2). Chegámos a resultados favoráveis de uma forma relativamente rápida, sendo que o veículo descreve elipses sem nunca se desviar da trajectória inicial.

Quanto ao segundo cenário (figura 3), onde o veículo deveria delinear um 8 em volta de dois focos, tivemos muitas mais dificuldades. Precisámos de muitas afinações e redefinições das posições iniciais dos blocos, do próprio veículo e parâmetros das funções de activação, mas foi com satisfação que conseguimos colocar o veículo em órbita.

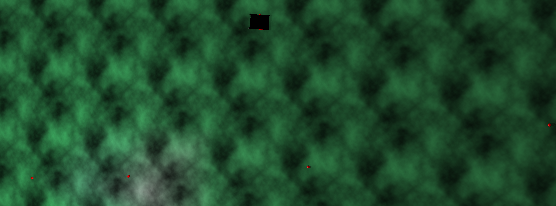
Para finalizar, definimos um cenário onde o nosso veículo ‘3c’ pudesse demonstrar as suas capacidades. Usamos todos os tipos de fontes (luz: vermelho, som: verde, olfacto: azul) e alguns blocos. De modo a facilitar a geração destes cenários, fizemos um pequeno pedaço de código que lê de um ficheiro de texto uma configuração em ASCII e a traduz para o efeito desejado. Para além disso, temos ainda um cenário para o veículo explorador, que procura sempre as luzes mais distantes, voltando para trás caso não tenha mais nenhuma à sua frente.

**Figura 1. Um cenário de exploração para o veículo 3c. Figura 2. O veículo realiza uma elipse.**



**Figura 3. Sequência de imagens do ‘circuito octal’ feito pelo veículo.**



**Figura 4. O veículo explorador procura as luzes mais distantes, voltando para trás se nada encontrar à sua frente.**

# Contributos Individuais

Para esta meta, torna-se mais difícil distinguir o trabalho feito por cada elemento do grupo, pois ambos contribuíram para todas as fases exigidas. Contudo, pode-se dizer que o aluno Ivo Correia contribuiu com maior relevância para a redacção do relatório e definição dos veículos Explorador e 3c (assim como os respectivos cenários de teste); enquanto o João Barbosa participou de forma mais activa na definição da classe de funções de activação e ajustamento dos parâmetros para que o veículo padrão estabelecesse uma elipse e um ‘circuito octal’ em torna de dois emissores.

Foram necessárias mais horas de trabalho para esta meta, especialmente porque foi gasto muito tempo na afinação dos parâmetros para obter os comportamentos desejados para cada veículo. Consequentemente, foram gastas aproximadamente 15 horas por cada aluno.

# Perspectivas Futuras

Tal como foi dito na introdução, sem bem que perdemos mais tempo nesta meta daquele que tínhamos inicialmente previsto, não foi todo um desvio improdutivo. Bem pelo contrário, descobrimos soluções para inúmeros problemas que se iriam levantar quando começássemos a implementar de uma forma mais séria e dedicada a última fase deste primeiro projecto. Especialmente a definição dos cenários de exploração permitiram-nos elaborar com mais precisão planos de um possível labirinto de modo a conter veículos no seu interior.

# Bibliografia

1. *Software Breve*
2. Documentação disponível em ‘*http://www.spiderland.org*’
3. Imagens: **Da capa:**
   1. <http://www.cyberbotics.com>

**Outras:**

* 1. *Screenshots* do ambiente *Breve* em execução para os cenários de teste.