



Introdução à Inteligência Artificial

(2015/2016 – 2º Semestre)

Trabalho Prático nº 1

Relatório

João André Taborda Portugal Vieira
nº 2013136370

joaoportugalv@gmail.com PL7

João Miguel dos Santos Faria
nº 2013136446

joaomdsfaria@hotmail.com PL7

João Guilherme Assafrão Craveiro
nº 2013136429

jgac@student.dei.uc.pt PL2



Índice

[Introdução](#)

[Meta 1](#)

[Meta 2](#)



Introdução

Este trabalho tem como objetivo a implementação de diversos veículos de braitenberg de maneira a adquirir conhecimentos acerca desta área e de competências relacionadas com o desenvolvimento e implementação de agentes autónomos

Meta 1

Nesta meta melhorámos o veículo já implementado pelo professor de maneira a que os sensores dele reconheçam luzes e blocos. No ambiente enviado é possível verificar que o veículo, ao andar pelo ambiente, vai atrás das luzes no seu ângulo de visão, desviando-se de blocos que apareçam no seu caminho (optámos por fazer com que o veículo se desvie sempre para a direita), sendo assim possível verificar que os sensores estão a trabalhar como desejado.

Durante a implementação fizemos várias alterações que afectaram os resultados na simulação. Para garantir que o veículo se desloca sempre que tem uma luz à frente, decidimos aumentar o raio dos sensores para um ângulo de 170°. Além disso, demos aos sensores de proximidade um raio de 20 para que o carro só se desvie dos blocos quando está perto deles.

O ambiente que apresentamos foi desenhado de maneira a que o veículo, ao desviar-se de blocos, se desloque na direcção de outra luz, resultando num movimento circular. Assim, podemos observar o comportamento do veículo durante algum tempo, até ele deixar de observar luzes à sua frente.

No código do script que controla o carro fornecido pelo professor, aumentámos ligeiramente a velocidade das rodas e fizemos com que a velocidade da roda esquerda fosse proporcional à proximidade com obstáculos e que a velocidade da roda direita fosse inversamente proporcional. Isto resulta no desvio de obstáculos pela direita que o carro apresenta.

No código do script do detector de proximidade, utilizamos um ciclo for que contido na função Update() para comparar a distância do veículo a cada bloco no seu ângulo de visão. Escolhemos considerar apenas o obstáculo mais próximo, e apenas se este estiver a menos de 20 unidades de distância do carro. Se não encontrar nenhum bloco, devolve um output 0, significando que o movimento do carro fica dependente apenas da quantidade de luzes no meio envolvente.



Meta 2

Nesta meta voltamos a alterar os sensores do carro de maneira a que possamos criar comportamentos mais interessantes. Para isso, implementámos uma função de activação guassiana, pois a linear(utilizado na meta 1) não permite criar estes comportamentos. O utilizador poderá no nosso programa, optar por usar a função guassiana ou a função linear, além disso também poderá também alterar os valores dos limites superiores e inferiores e os limiares (Figura 1) .

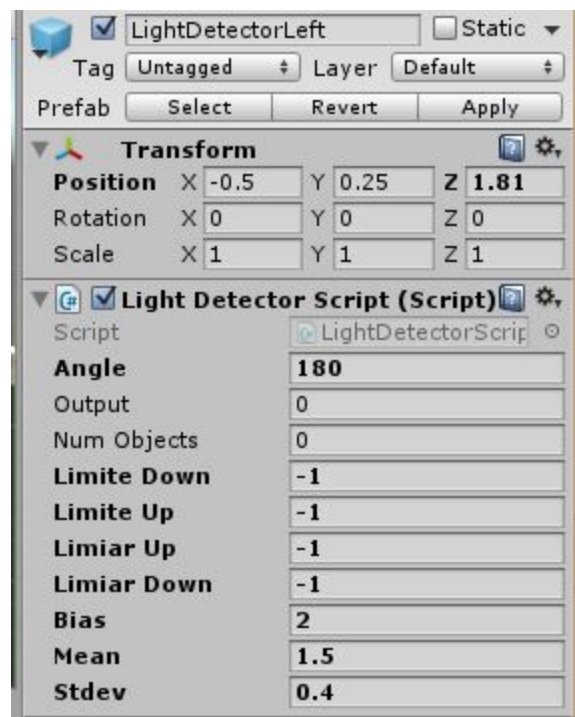


Figura 1: Menu dos sensores onde é possível especificar várias opções



Usando a função de activação guassina, conseguimos criar vários comportamentos únicos, que não era possível obter na meta 1 devido as restrições criadas pela função linear. Assim foi possível replicar as trajetórias pedidas

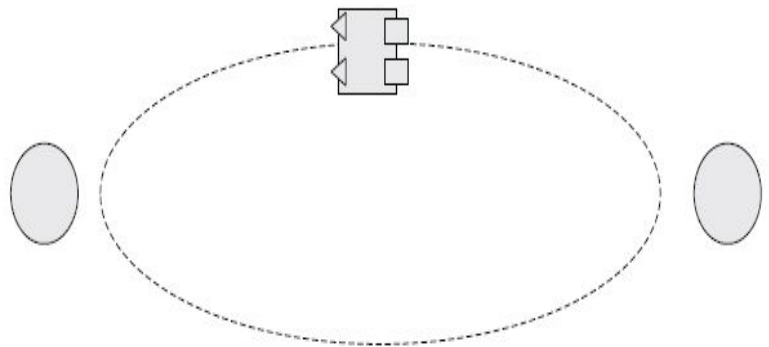


Figura 2: Trajetória em eclipse

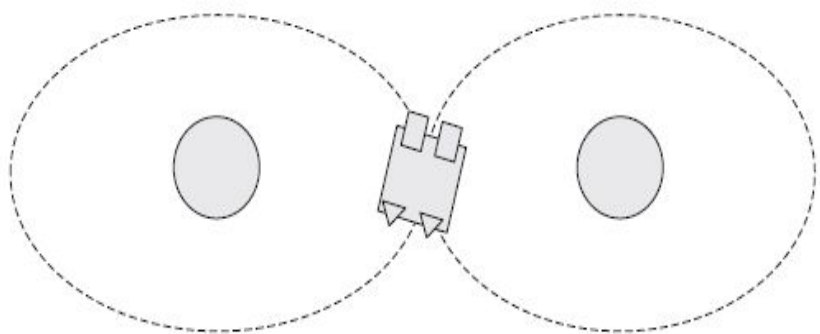
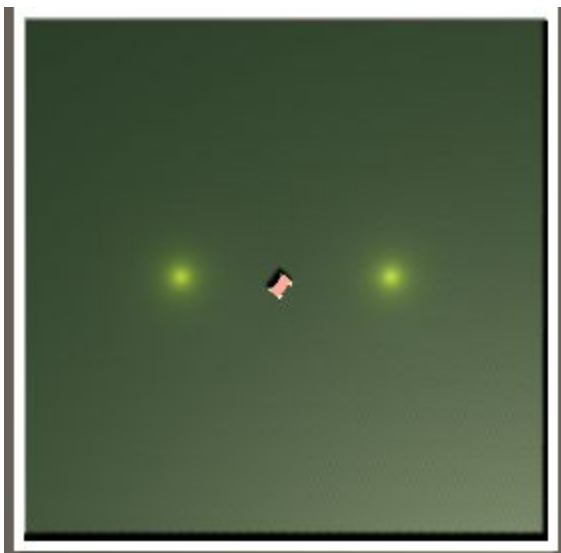


Figura 3: Trajetória em oito.