



UNIVERSIDADE DE  
COIMBRA

FACULDADE  
DE CIÊNCIAS  
E TECNOLOGIA

Licenciatura em Engenharia Informática

Introdução à Inteligência Artificial

2019/2020 – 2º Semestre

Reactive D31: The AI Awakens

*Diogo Jorge da Rocha Cota*

2017267601

[uc2017267601@student.uc.pt](mailto:uc2017267601@student.uc.pt) PL4

*João Filipe Carnide de Jesus Nunes*

2017247442

[uc2017247442@student.uc.pt](mailto:uc2017247442@student.uc.pt) PL4

*Raul Filipe Enes Nogueira*

2017267634

[uc2017267634@student.uc.pt](mailto:uc2017267634@student.uc.pt) PL4

## Índice

Introdução	2
Meta 1 – Sense It	3
Objetivos Alcançados	3
Dificuldades	3
Meta 2 – Tune It & Test It	4
Objetivos Alcançados	4
Map1a	4
Map1b	5
Map2a	6
Map2b	7
Conclusão	9

## Introdução

Neste primeiro trabalho prático pretende-se implementar agentes reativos que, através de sensores, consiga, num certo ambiente, desviar-se de certos obstáculos e identificar recursos para os colecionar. Ou seja, pretende-se que o *Reactive D31* seja um agente reativo completamente autónomo, não necessitando de ajuda de terceiros para alcançar os seus objetivos.

Na meta 1 foi pedido a implementação dos dois sensores necessários para que o agente conseguisse circular nos respetivos mapas, um sensor para detetar recursos, colecionando-os, e um para detetar blocos, evitando-os.

Na meta 2, como base nos sensores implementados na meta 1, foi pedido para implementar limiares e limites no que diz respeito às forças e energias, respetivamente, que atuam face ao agente, blocos e recursos. Com base nisto foram testados os comportamentos do agente nos diferentes mapas.

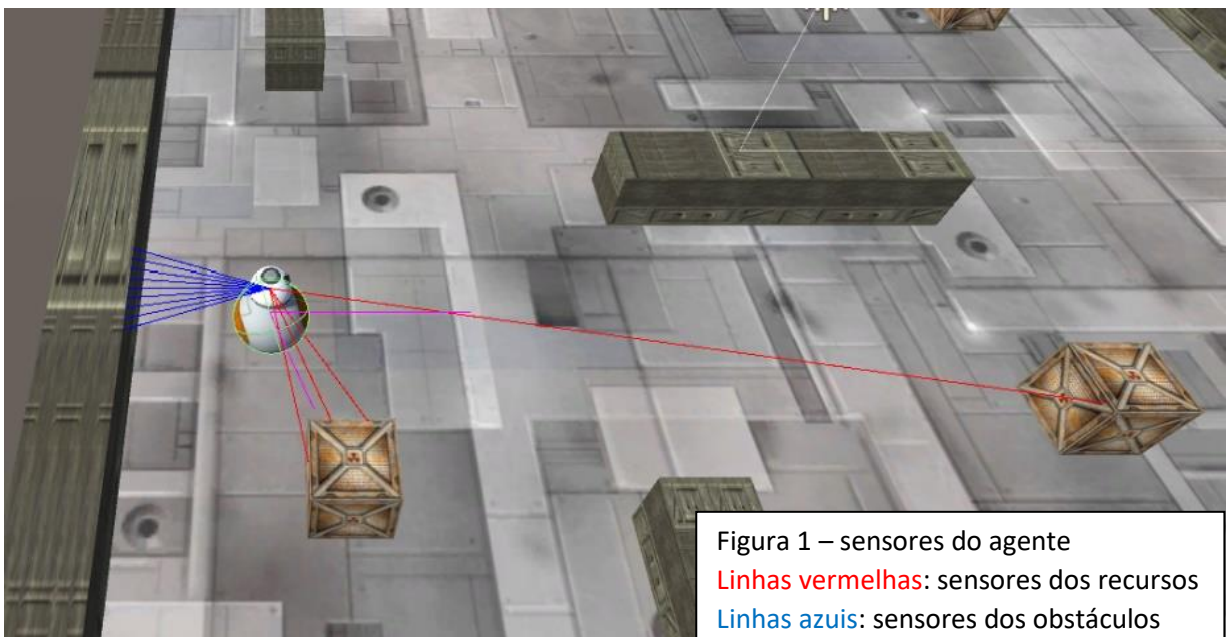
# Meta 1 – Sense It

## Objetivos Alcançados

Os objetivos para esta meta eram a implementação de sensores para o agente que fizesse com que este colecionasse os recursos (caixas) e evitasse os obstáculos que estivessem no seu caminho.

Os sensores de reconhecimento dos recursos já estavam previamente implementados, por isso, coube-nos apenas a implementação dos sensores para que o agente *Reactive D31* evitasse os obstáculos no seu caminho. Esta implementação foi baseada na dos sensores de reconhecimento dos recursos, contudo como pretendemos que o agente se afaste dos obstáculos, o ângulo dos blocos foi aumentado 180 graus, assim o agente quando vê os obstáculos tenta ao máximo afastar-se deles.

Também para os sensores funcionarem todos corretamente e o agente não colidir com os obstáculos foram mudadas variáveis como a velocidade e os pesos, tanto dos recursos como dos obstáculos.



## Dificuldades

Nesta meta a principal dificuldade que o grupo sentiu foi estar a mexer constantemente nos pesos dos recursos e dos obstáculos, pois em vários casos funcionava sem que o agente colidisse com os obstáculos e em outros a aceleração com que vinha da recolha de um recurso era tão grande que iria colidir com as paredes do mapa, principalmente nas do map1b.

## Meta 2 – Tune It & Test It

### Objetivos Alcançados

Os objetivos nesta meta foram alcançados com base em diversos testes realizados nos vários mapas com base nas combinações dos sensores (recursos e blocos) e das funções de ativação (linear, gaussiana e logarítmica). Os resultados também tiveram como base os limiares (*threshold*) mínimo e máximo e os limites superiores e inferiores.

Quanto às funções de ativação implementadas as fórmulas utilizadas foram as seguintes:

- Função Linear:  $y = x$ , sendo  $x$  a força e  $y$  a energia
- Função Gaussiana:  $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$ , onde  $x$  é a força,  $f(x)$  a energia,  $\mu$  a média e  $\sigma$  o desvio padrão, onde estes dois últimos apresentam valores variáveis nos vários mapas.
- Função Logaritmo:  $y = 1/\log(x)$ , onde  $x$  representa a força e  $y$  a energia

No que diz respeito aos resultados obtidos foram feitas diversas medições de tempo para os diferentes mapas com combinações das funções de ativação, onde se chegou a uma conclusão de qual a melhor combinação para os diferentes mapas.

### Map1a

Neste mapa foram utilizados os limiares e os limites presentes no enunciado, ou seja, limiar inferior de 0.25, limiar superior de 0.75, limite inferior de 0.05 e limite superior de 0.6. Na função de ativação Gaussiana os valores da média e do desvio padrão também foram os que estão presentes no enunciado ( $\mu = 0.5$  e  $\sigma = 0.12$ ).

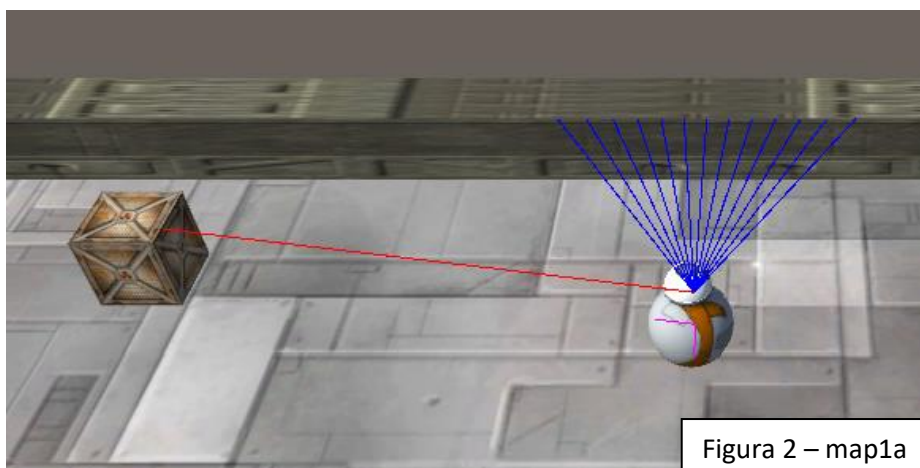


Figura 2 – map1a

### Tabela de tempos

	Recursos Linear Blocos Linear	Recursos Linear Blocos Gaussiana	Recursos Linear Blocos Logaritmo	Recursos Gaussiana Blocos Linear	Recursos Gaussiana Blocos Gaussiana	Recursos Gaussiana Blocos Logaritmo	Recursos Logaritmo Blocos Linear	Recursos Logaritmo Blocos Gaussiana	Recursos Logaritmo Blocos Logaritmo
<b>Sem Pesos</b>	22s	20s	30s*	26s	25s	30s*	47s	26s	Parou
	22s	25s	44s*	Parou	17s	30s*	25s	25s	28s
	24s	24s	38s*	24s	27s	36s*	26s	23s	Parou
<b>Com Pesos</b>	20s	22s	24s*	24s	Parou	24s*	24s	22s	26s
	30s	23s	26s*	29s	20s	29s*	24s	27s	22s
	18s	22s	28s*	25s	24s	25s*	23s	25s	22s

\*bateu às vezes nas paredes

### Conclusões

As conclusões que tiramos da análise da tabela de tempo do map1a é que podemos excluir 5 combinações pois não são consistentes, ou porque bateram às vezes nas paredes ou porque o agente parou a meio do seu percurso devido ao facto do sensor não conseguir detetar os recursos a uma certa distância.

Na comparação das combinações restantes foi feita a média dos 6 tempos medidos e chegamos à conclusão que existem duas combinações melhores que as outras, tanto a combinação recursos linear/blocos linear como a combinação recursos linear/blocos gaussiana apresentam a melhor média dos tempos medidos (22.7 segundos). Agora, para a distinção entre estas duas combinações, foi feita a média para quando possuía pesos e quando não.

Finalmente, a conclusão chegada foi que, para este mapa, a combinação recursos linear/blocos linear é a melhor para quando não existem pesos e a combinação recursos linear/blocos gaussiana a melhor para quando existem pesos.

### **Map1b**

Neste mapa foram utilizados os limiares e os limites presentes no enunciado, ou seja, limiar inferior de 0.25, limiar superior de 0.75, limite inferior de 0.05 e limite superior de 0.6. Na função de ativação Gaussiana os valores da média e do desvio padrão também foram os que estão presentes no enunciado ( $\mu = 0.5$  e  $\sigma = 0.12$ ).

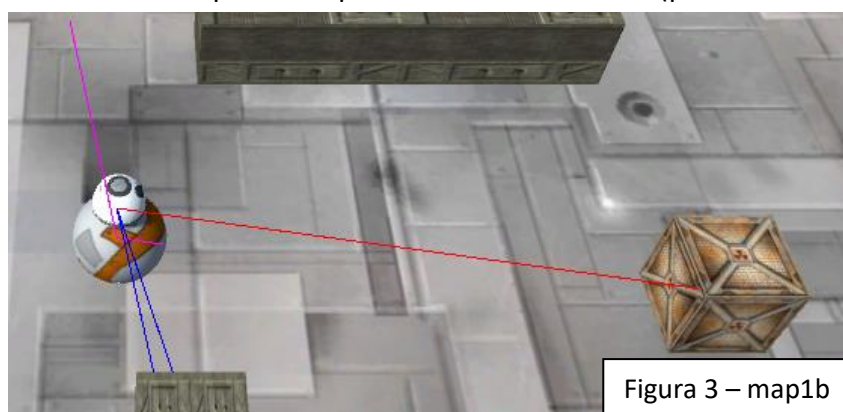


Figura 3 – map1b

Tabela de tempos

	Recursos Linear Blocos Linear	Recursos Linear Blocos Gaussiana	Recursos Linear Blocos Logaritmo	Recursos Gaussiana Blocos Linear	Recursos Gaussiana Blocos Gaussiana	Recursos Gaussiana Blocos Logaritmo	Recursos Logaritmo Blocos Linear	Recursos Logaritmo Blocos Gaussiana	Recursos Logaritmo Blocos Logaritmo
<b>Sem Pesos</b>	30s* 21s 59s	23s 44s 89s	20s* 20s* 20s*	Parou 39s 27s	Parou 39s 24s	20s* 20s* 20s*	42s 51s 46s	37s 27s 33s	39s 43s 38s
<b>Com Pesos</b>	32s* 25s* 30s*	67s 23s 33s	19s* 20s* 20s*	Parou Parou 21s*	Parou Parou 23s	20s* 20s* 20s*	44s 32s 45s	45s 34s 33s	37s 33s 43s

\*bateu às vezes nas paredes

### Conclusões

Analisou-se os resultados da tabela e verificou-se que os casos em que o agente completa o mapa com sucesso, isto é, sem bater nas paredes, são nas combinações recursos linear/blocos gaussiana, recursos logaritmo/blocos linear, recursos logaritmo/blocos gaussiana e recursos logaritmos/blocos logaritmo.

A partir destes casos de sucesso foi feita uma média dos tempos atingidos pelo agente em cada caso. Observou-se que, sem pesos, a melhor combinação é a recursos logaritmo/blocos gaussiana (32s). Com pesos, as combinações recursos logaritmo/blocos gaussiana e recursos logaritmo/blocos logaritmos revelaram-se as melhores combinações com uma média de 37.3s e 37.6s, respetivamente.

Concluiu-se que a combinação recursos logaritmo/blocos gaussiana resolve o mapa com melhor eficácia e rapidez do que as restantes combinações.

### **Map2a**

Neste mapa foram utilizados maioritariamente os limiares e os limites presentes no enunciado, ou seja, limiar inferior de 0.25, limiar superior de 0.75 e limite superior de 0.6, contudo, o limite inferior foi reduzido para 0.005. Na função de ativação Gaussiana os valores da média e do desvio padrão também foram os que estão presentes no enunciado ( $\mu = 0.5$  e  $\sigma = 0.12$ ).

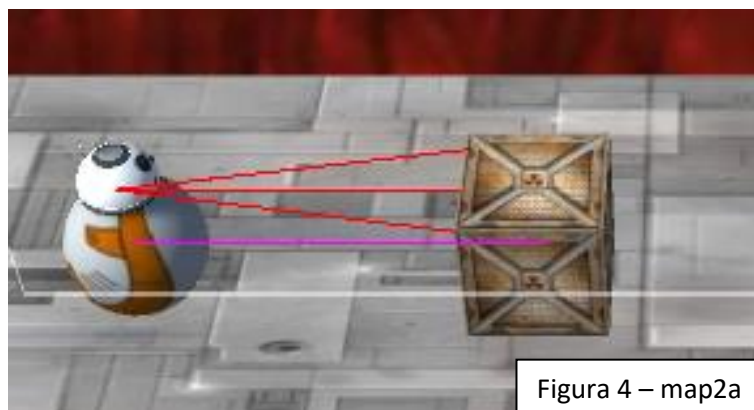


Figura 4 – map2a

Visto que o tempo para o agente colecionar o único recurso presente no mapa é equivalente em todos os casos, as medições deste mapa foram feitas através da verificação se o agente caia na lava ou não.

#### Tabela de verificação

	Recursos Linear	Recursos Gaussiana	Recursos Logaritmo
<b>Sem Pesos</b>	Caiu Caiu Caiu	Caiu Caiu Caiu	Parou Parou Parou
<b>Com Pesos</b>	Caiu Caiu Caiu	Caiu Caiu Caiu	Parou Parou Parou

#### Conclusões

Analisando as execuções no map2a concluímos que poderíamos ignorar a força aplicada perante a presença de blocos devido à ausência do mesmo.

Estudando todos os casos possíveis para a força aplicada perante a presença de recursos concluímos que o melhor e único caso em que deverá funcionar, sem o agente cair, é quando lhe é aplicada uma força pela função logaritmo, pois este faz com que o agente ao aproximar-se do recurso vá diminuindo a velocidade, ao contrário dos outros casos que ao aproximarem-se do recurso aumentam a força aplicada o que leva à queda do agente.

Ver vídeo do comportamento do agente com a função logaritmo sem a atuação de pesos em <https://youtu.be/r92ifn3Bums>.

#### **Map2b**

Neste mapa foram utilizados os limiares e os limites presentes no enunciado, ou seja, limiar inferior de 0.25, limiar superior de 0.75, limite inferior de 0.05 e limite superior de 0.6. Na função de ativação Gaussiana os valores da média e do desvio padrão também foram os que estão presentes no enunciado ( $\mu = 0.5$  e  $\sigma = 0.12$ ).

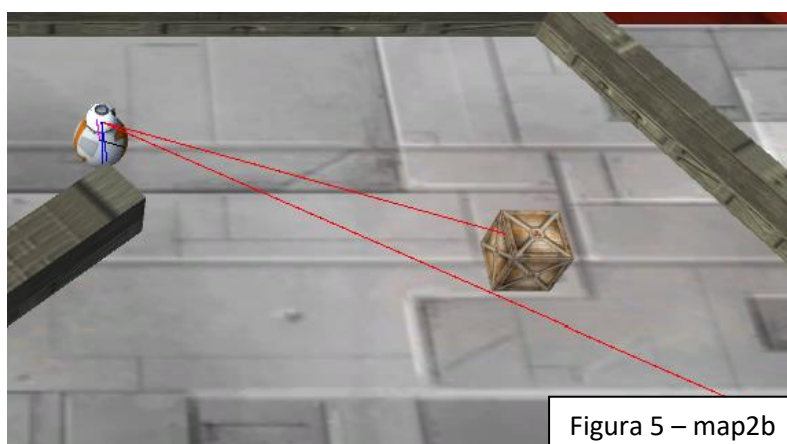


Figura 5 – map2b



Tabela de tempos

	Recursos Linear Blocos Linear	Recursos Linear Blocos Gaussiana	Recursos Linear Blocos Logaritmo	Recursos Gaussiana Blocos Linear	Recursos Gaussiana Blocos Gaussiana	Recursos Gaussiana Blocos Logaritmo	Recursos Logaritmo Blocos Linear	Recursos Logaritmo Blocos Gaussiana	Recursos Logaritmo Blocos Logaritmo
<b>Sem Pesos</b>	Caiu Caiu Caiu	Parou Parou Caiu	Caiu Caiu Caiu	Caiu 59s* Caiu	Caiu 90s* 69s*	Caiu Caiu Caiu	158s 87s Parou	Parou 168s Caiu	Parou Parou Parou
<b>Com Pesos</b>	Parou Caiu Parou	Caiu Caiu Caiu	Caiu Parou Caiu	Caiu Caiu Caiu	Caiu Caiu Caiu	Caiu Caiu 29s*	77s Caiu Parou	Caiu 165s 172s	Parou Parou Parou

\*bateu às vezes nas paredes

### Conclusões

Estudando as combinações possíveis para o map2b, verificamos que se a força de aproximação aos recursos dos recursos for baixa o agente conseguirá (na maioria dos casos) evitar a queda, pois a sua força irá baixar ou, caso seja possível ver outro recurso, irá mudar de direção para se aproximar do mesmo, logo conclui-se que o melhor algoritmo seria o negativo do logaritmo pois é o único em que a força diminui.

Em relação à força aplicada em relação aos blocos, esta deverá ser mais forte que a força dos recursos, pois, caso o recurso esteja muito próximo de um bloco, esta força consiga contrariar a força dos recursos para que o agente não choque com nenhum bloco. Na teoria os algoritmos linear e gaussiana conseguem satisfazer estas condições pois são mais fortes que o negativo do logaritmo, mas depois de analisarmos os resultados dos nossos testes, verificamos que o robô em certas alturas caía, por a força que lhe era aplicada para evitar os blocos o levar para um local em que este poderia cair, ou parava por não conseguir detectar mais nenhum recurso, uma das razões deste acontecimento era o caso em que a força de evitar os blocos levava o agente para um local em que este não conseguiria detectar nenhum recurso.

Conclui-se, então, que a melhor combinação será a força dos recursos ser gerada pelo negativo do logaritmo, e a força dos blocos pela gaussiana ou pela função linear.

Ver vídeo do comportamento do agente com a função logaritmo nos recursos e função linear nos blocos sem a atuação de pesos em <https://youtu.be/XGgZehhgGZM>.



## Conclusão

Em suma, este trabalho prático permitiu a implementação de dois tipos de sensores, um de desvio de obstáculos e outro de aproximação de recursos, num agente de modo a que este possa agir o mais independentemente possível, sem necessitar de terceiros para efetuar o seu objetivo, colecionar todos os recursos que estão presentes num mapa.

No diz respeito aos resultados obtidos, com as diferentes combinações utilizadas nos diferentes ambientes, chegámos à conclusão de que a função de ativação logaritmo é ideal para os recursos pois apresenta uma força controlada, esta força vai diminuindo até o agente colecionar os recursos. No que diz respeito à função de ativação para os blocos chegámos à conclusão de que a função gaussiana é uma boa opção pois a força é máxima quando estiver o mais próximo do bloco, e vai diminuindo à medida que afasta, impedindo assim o agente que colida com os obstáculos.

Com estas funções de ativação, o agente conseguiu uma deslocação mais segura ao longo do seu percurso num determinado ambiente sem que colidisse com os obstáculos e que apanhasse todos os recursos presentes num mapa.