t

Licenciatura em Engenharia Informática

# Índice

[Índice 3](#_Toc87634749)

[Introdução 4](#_Toc87634750)

[Implementação 5](#_Toc87634751)

[Modelo Base: 5](#_Toc87634752)

[Modelo Melhorado: 6](#_Toc87634753)

[Ninjas 6](#_Toc87634754)

[Células Explosivas (BOOOM!) 6](#_Toc87634755)

[Análise de Resultados 7](#_Toc87634756)

[Modelo Base: 7](#_Toc87634757)

[Alterar a energia que os agentes ganham ao comer 7](#_Toc87634758)

[Alterar a energia inicial dos agentes 7](#_Toc87634759)

[Alterar o número de Limpadores 8](#_Toc87634760)

[Modelo Melhorado: 8](#_Toc87634761)

[Alterar o número de células explosivas 8](#_Toc87634762)

[Alterar o número de ninjas existentes 9](#_Toc87634763)

[Alterar o raio de explosão das células explosivas 9](#_Toc87634764)

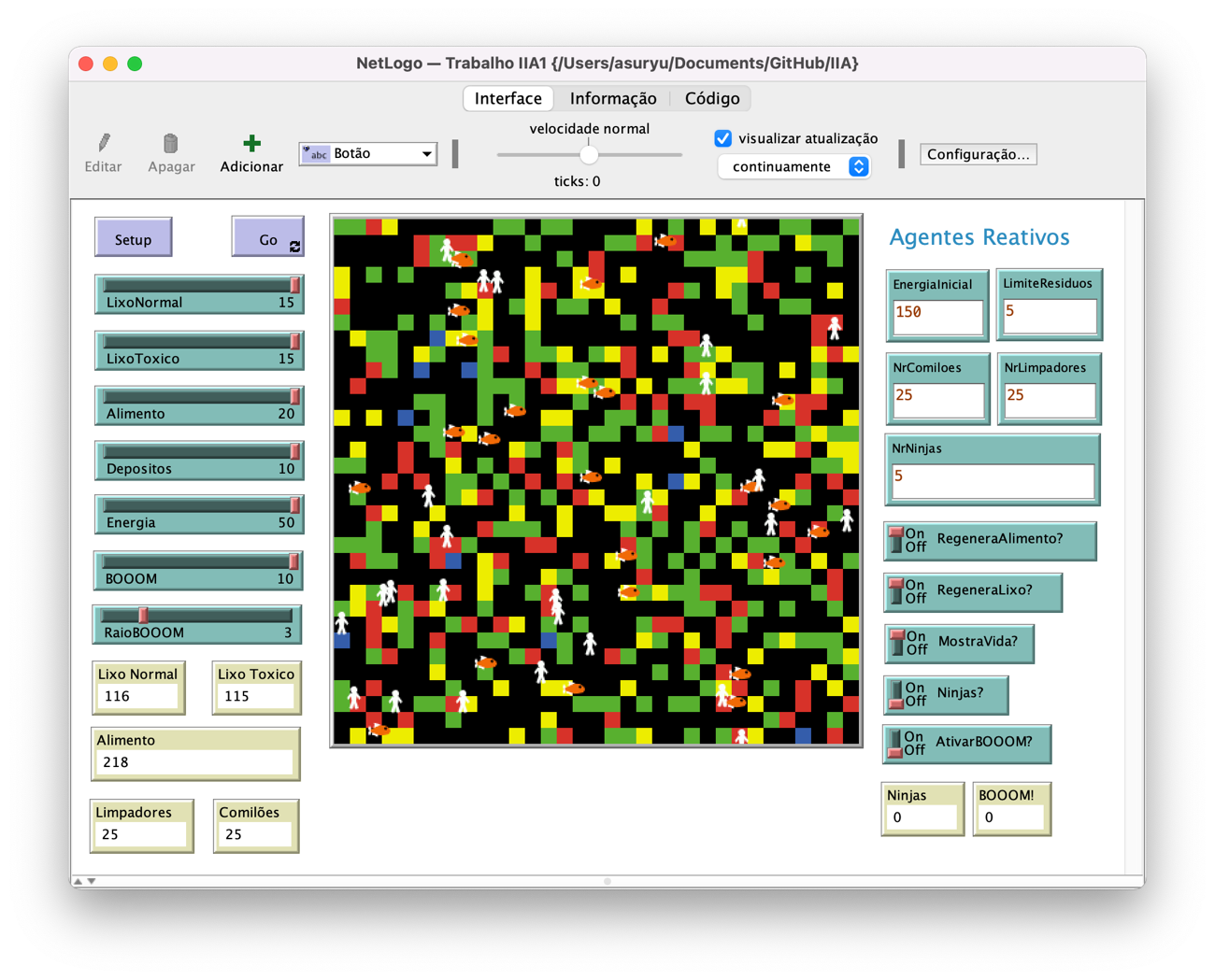
[Conclusão 10](#_Toc87634765)

# Introdução

O primeiro trabalho prático de Introdução à Inteligência Artificial consiste na criação de uma simulação com agentes reativos.

Foram criados dois modelos distintos: o **modelo base**, onde implementámos todas as características que foram pedidas e o **modelo melhorado**, onde criámos duas novas funcionalidades de modo a alterar o desempenho dos agentes.

Com ambos os modelos criados realizámos também três experiências para cada modelo para podermos olhar para os dados e perceber o que é que muda consoante os parâmetros que alteramos.



# Implementação

## Modelo Base:

No modelo base implementámos a criação do ambiente da simulação que consistiu na criação dos agentes e preenchimento do tabuleiro com *patches* de cores variadas correspondentes aos vários elementos (comida, lixo, etc..).

Uma imagem com texto

Descrição gerada automaticamente Uma imagem com texto

Descrição gerada automaticamente Uma imagem com texto

Descrição gerada automaticamente

Para além disso, criámos também procedimentos para controlar o comportamento dos Comilões e dos Limpadores.

Uma imagem com texto

Descrição gerada automaticamente

O procedimento **Go** tem como função avançar um tick e retirar uma unidade de vida a todos os agentes. Já o procedimento **VerificaMorte** trata de matar os agentes que tiverem uma vida menor ou igual a zero.

## Modelo Melhorado:

No que toca ao modelo melhorado, o nosso grupo de trabalho decidiu criar duas novas funcionalidades:

* Ninjas
* Células Explosivas (BOOOM!)

### Ninjas

Os **Ninjas** são agentes que aparecem para destabilizar o normal funcionamento da simulação. O trabalho destes agentes é encontrar alimentos e transformá-los em lixo normal (chance 80%) ou lixo tóxico (chance 20%).

Se os ninjas forem vistos por algum Comilão (ser visto significa estar ao lado de um agente) perdem energia equivalente à energia inicial a dividir por 10.

O número de Ninjas é configurável pelo utilizador com qualquer valor.

Uma imagem com shoji, palavras cruzadas, edifício

Descrição gerada automaticamenteUma imagem com shoji, palavras cruzadas, edifício

Descrição gerada automaticamente

Figura - Perceção dos Ninjas

### Células Explosivas (BOOOM!)

As **Células Explosivas (BOOOM!)** são células roxas que quando são calcadas por qualquer agente explodem todo o lixo ao redor da célula num raio configurável. Todos os agentes menos os ninjas têm perceções programadas para este tipo de células de acordo com as suas próprias perceções.

O número de células existentes e o raio de explosão é configurável com valores entre 1 e 10.

Uma imagem com texto

Descrição gerada automaticamente

Para criar estas células definimos um contador que vai correr até ao número de células configuradas pelo utilizador na interface e usamos o ***ask one-of patches*** para selecionar uma célula aleatória para a pintarmos com a cor roxa.

# Análise de Resultados

## Modelo Base:

No modelo base efetuámos 10 repetições (com condição de paragem nos 2500 ticks) de várias experiências com parâmetros alterados para testar a capacidade de sobrevivência dos agentes.

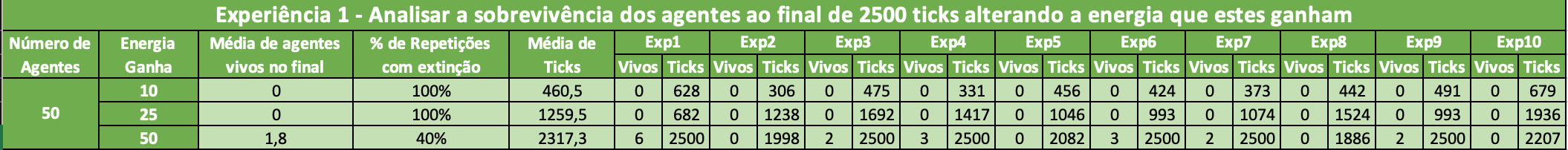
Os testes relativos ao modelo base são:

* Alterar a energia que os agentes ganham ao comer
* Alterar a energia inicial dos agentes
* Alterar o número de Limpadores

### Alterar a energia que os agentes ganham ao comer

Esta experiência consiste em analisar se ao fim de 2500 ticks sobrevivem mais agentes consoante a energia que eles ganham ao consumir um alimento. Os valores de energia que utilizámos são 10, 25 e 50.

**Será que aumentar a energia que os agentes ganham ao comer um alimento vai ter um impacto grande na capacidade de sobrevivência dos agentes?**

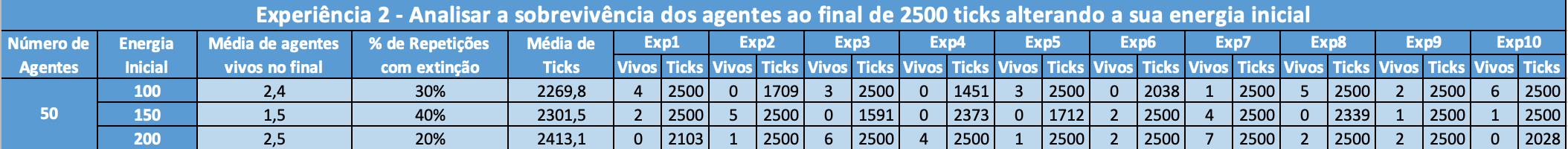


Olhando para os dados que se encontram na tabela, podemos reparar que: com mais energia ganha por alimento, os agentes aguentam-se mais tempo e aumenta também a chance de ao fim dos 2500 ticks ainda estarem vivos.

### Alterar a energia inicial dos agentes

Desta vez, experimentámos alterar os valores de energia com que cada um dos agentes começa. Os valores de energia inicial que utilizámos são 100, 150 e 200.

**Será que aumentar a energia inicial dos agentes vai permitir que estes sobrevivam mais tempo?**



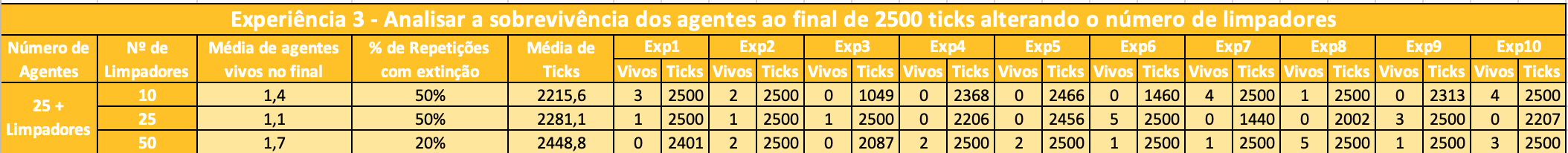
Como podemos observar, alterar a energia inicial não afeta a capacidade de sobrevivência dos agentes.

### Alterar o número de Limpadores

A última experiência do modelo base foca-se em alterar o número de Limpadores para averiguar se estes têm um impacto grande na sobrevivência dos agentes como um todo.

Testámos isto com 10, 25 e 50 Limpadores.

**Será que aumentar o número de Limpadores vai ajudar a recolher o lixo mais rápido e consequentemente aumentar a capacidade de sobrevivência dos agentes?**



Apesar dos Limpadores terem uma tarefa importante de limpar o lixo do mapa, parece que isso não afeta muito a quantidade de agentes que sobrevive ao fim de 2500 ticks.

Porém, com 50 Limpadores houve menos repetições com extinção total dos agentes pelo que se pode afirmar que estes amiguinhos que tratam de limpar as ruas da nossa simulação não são assim tão inúteis!

## Modelo Melhorado:

No modelo melhorado quisemos testar as nossas implementações originais como as células explosivas e os ninjas. Para isso, efetuámos 10 iterações de várias experiências de modo a observar a sobrevivência dos agentes ao fim de 2500 ticks.

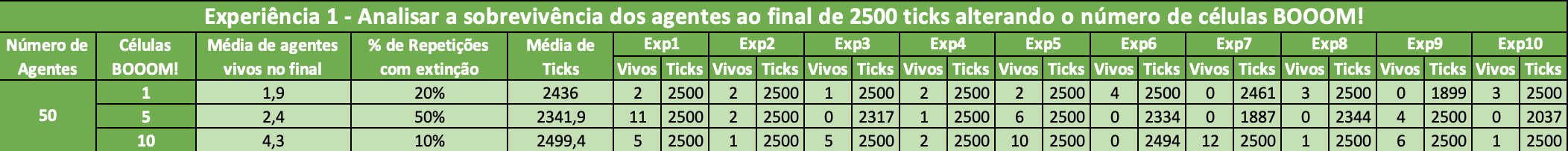
Os testes relativos ao modelo melhorado são:

* Alterar o número de células explosivas
* Alterar o número de ninjas existentes
* Alterar o raio de explosão das células explosivas

### Alterar o número de células explosivas

Na primeira experiência do modelo melhorado começámos por alterar o número de células explosivas existentes no mapa. Na interface temos um *slider* que nos permite alterar este valor entre 1 e 10 e nesta experiência. Para esta experiência em concreto, colocámos espalhadas no mapa 1, 5 e 10 células explosivas.

**Será que aumentar o número de células explosivas vai ajudar os agentes a sobreviverem mais tempo?**



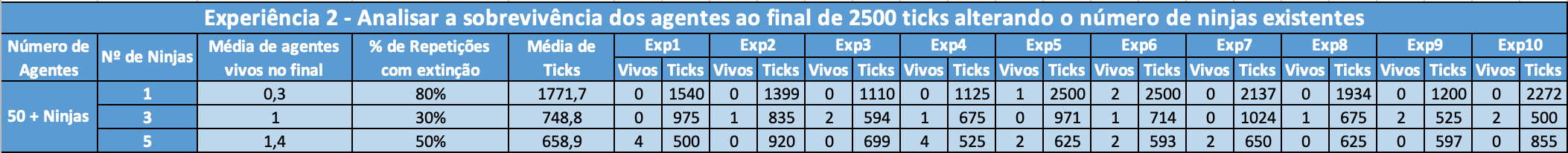
Como as células explosivas limpam apenas o lixo num raio predefinido isto ajuda os agentes a não morrerem para o lixo. Isto explica o facto da média de agentes vivos no final dos 2500 venha aumentar com mais células explosivas.

### Alterar o número de ninjas existentes

Uma das nossas implementações para o método melhorado foi a criação de ninjas capazes de roubar a comida e trocá-la por um dos dois tipos de lixo.

Neste teste, experimentámos colocar 1, 3 e 5 ninjas ativos na simulação para observar a capacidade de sobrevivência dos restantes agentes mesmo com os ninjas a destabilizar.

**Será que aumentar o número de ninjas existentes vai atrapalhar os restantes agentes de modo que estes sobrevivam menos tempo?**



Estranhamente, a média de agentes vivos no final de cada iteração não ficou muito abalada com a presença de mais ninjas, mas algo que é bastante notório é a reduzida quantidade de ticks necessários para que todos os agentes morram. Ou seja, com mais ninjas os agentes aguentam, em média, menos tempo na simulação.

### Alterar o raio de explosão das células explosivas

Para a nossa última experiência, quisemos testar algo relacionado com a funcionalidade do modelo melhorado em que mais gostámos de trabalhar: as células explosivas. A primeira experiência deste modelo também era relacionada com este tipo de células, mas, enquanto essa dizia respeito a número de células explosivas espalhadas no mapa, esta foca-se no raio de explosão destas mesmas células.

Experimentámos com um raio de explosão de 3, 5 e 10 unidades, respetivamente.

**Será que aumentar o raio de explosão das células explosivas vai fazer diferença no que toca à sobrevivência dos agentes?**



Como era de esperar, nota-se alguma diferença na capacidade de sobrevivência dos agentes visto que existe menos lixo para “atrapalhar”. Com um aumento do raio de explosão verificamos um aumento da média de agentes vivos no final de cada simulação.

Algo a notar é que não houve nenhuma repetição com extinção pois limitámos o número de ticks máximo em 2500. Se a simulações tivessem corrido mais tempo poderíamos ter obtido um número mais preciso.

# Conclusão

Este trabalho permitiu-nos consolidar bem a matéria das aulas de Introdução à Inteligência Artificial relativas ao **NetLogo** e dessa forma pudemos colocar-nos à prova relativamente àquilo que realmente sabíamos e também pudemos aplicar os nossos conhecimentos.

Por se tratar de uma simulação com um grande carácter visual e que de certa forma se assemelha a um jogo, creio que isso nos deu uma grande motivação para realizarmos este trabalho prático.

