Instituto Superior de Engenharia de Coimbra

Introdução à Inteligência Artificial

Trabalho Prático 1

Agentes Racionais

Licenciatura em Engenharia Informática

2024 / 2025

Nuno Tomás Paiva

a2023137363@isec.pt

Rui Martins dos Santos

a2023145822@isec.pt

Índice

1. Introdução

O objetivo deste trabalho é introduzir a matéria inicial da Unidade Curricular de Introdução à Inteligência Artificial para os alunos começarem a experimentar e testar as funcionalidades do Netlogo.

Achamos que este trabalho tem muita importância dado que, ainda vamos fazer um trabalho de maior cotação da Unidade Curricular mais para a frente do semestre.

Dado que os Agentes têm um papel importante na matéria da UC é importante fazer um trabalho que consiste em conceber, implementar e analisar comportamentos racionais em agentes reativos.

2. O Ambiente

O ambiente é delimitado por uma grelha bidimensional não toroidal ou fechada. O "fundo" são células pretas, enquanto que, as células vermelhas são o lixo (a quantidade de células que o ambiente contém é configurável pelo utilizador). Assim que um agente passa na célula vermelha significa que a célula foi aspirada e então muda para a cor preta. Ainda existem mais 3 cores diferentes nas células, azuis, verdes e brancas. As azuis são carregadores para o aspirador enquanto que as verdes são depósitos de lixo. As células brancas são obstáculos ou aspiradores sem energia (que morreram), e os aspiradores com energia devem se desviar desses obstáculos. A quantidade de células brancas e azuis iniciais, também é definida pelo utilizador.

2.1 Funções

Setup – Serve para criar o modelo, chama as funções setup-patches e setup-turtles.

Setup-patches – Serve para criar as patches, colocando aleatoriamente no tabuleiro patches vermelhas com a função random : "if random 101 <= percentagemVermelho (definido pelo utilizador)". O tamanho de cada patch é definido por "set-patch-size". Para fazer o depósito utilizamos uma variável (flag) para saber quando existirem quatro células vizinhas pretas para ficarem de cor verde.

Setup-turtles – Para criar os aspiradores utilizamos: "create-aspiradores naspiradores". O "set shape "target" " serve para definir o aspeto dos nossos agentes. Colocamos o lixo inicial a 0, a direção para 0 graus e a energia sendo a energia definida pelo utilizador. A capacidade de carga é definida pelo utlizador também. Definimos as coordenadas do depósito e do carregador para 1000 para quando souber de facto onde estão os carregadores ou o depósito atualizar por cima desses valores. Os ticks de tempo de espera de carregamento e de despejar o lixo são ambos definidos pelo utilizador.

3. Os Agentes

Os agentes no nosso trabalho são os aspiradores e a sua quantidade é definida pelo utilizador, juntamente com a sua quantidade inicial de energia e a capacidade de carga.

O objetivo inicial dos agentes deverá ser sempre apanhar o lixo presente no modelo, esta ação só será possível caso o aspirador tenha energia. Assim que o aspirador entrar no modo poupança, irá ignorar o lixo por onde passar focando-se assim em encontrar um posto de carregamento. Voltando aí a estar com a energia completa, e depois, o aspirador retorna ao objetivo inicial de apanhar lixo.

A mesma situação acontece com a capacidade de carga, caso o aspirador não tenha mais capacidade, ele irá deixar de coletar o lixo e passará a focar-se na procura de um depósito de lixo.

4. Modelo Base vs Modelo Melhorado

Em diferença com o Modelo Base, o Modelo melhorado passa a informação às suas percepções de vizinhança sobre a local do deposito de lixo, memorizam o local do deposito de lixo, tem sensores para quando as suas percepções de vizinhança detetarem: algum lixo (limparem-no), local de carregamento (memorizarem-no) e deposito de lixo (memorizarem-no), implementámos uma reprodução, quando passa em cima de um depósito, mesmo que não esteja na capacidade máxima de lixo, retirar o lixo (depositá-lo). Fizemos, também, um lixo que demora mais tempo a ser limpo, o aspirador memoriza não apenas uma localização de um carregador, mas várias e regressa à base de carregamento mais próxima e, implementamos uma condição em que os sensores se estragam (os sensores deixam de funcionar) depois de depositar lixo um certo número de vezes. Por fim, quando já não existe lixo, no Modelo Melhorado, os aspiradores vão depositar o lixo ao deposito, enquanto no Modelo Base, os aspiradores ficavam com o lixo neles.

4.1 Funções do Modelo Base

Go – Função responsável por colocar o programa a funcionar, clicando no botão "Go" e usando as funções "MoveAspiradores" e "InfoNeighbour", para além de parar o programa caso não haja mais aspiradores "vivos".

LimitesModelo – Função responsável para os aspiradores respeitarem os limites do tabuleiro e rodarem 180º graus caso tenham chegado ao limite.

RandomSpawn – Função que gera os aspiradores aleatoriamente no tabuleiro

AheadWhite – Nesta função, os aspiradores ao depararem-se com uma patch branca à frente, têm 50% de probabilidade de virarem-se para esquerda e 50% de probabilidade de virarem-se para a direita.

AheadGeral – Função básica, a sua única função é fazer com que o aspirador perca 1 ponto de energia sempre que anda uma patch.

StopWaitingCarregamento – Caso o aspirador esteje em carregamento ele vai decrementando o número de ticks que o utilizador definiu.

StopWaitingLixo – Situação igual à função StopWaitingCarregamento

MemorizaCarregador – O aspirador começa com as coordenadas iniciais com 1000 nos dois paramentros(x e y) porém assim que encontra um carregador, decora as coordenadas desse mesmo carregador.

AheadCarregar – Função que direciona o aspirador para o carregador que tem as coordenadas na memória.

InfoNeighbour – Função responsável por dar as coordenadas do carregador e do depósito aos aspiradores que passarem na vizinhança de um aspirador conhecedor das coordenadas.

MoveAspiradores – Função principal, chama todas as outras funções, verifica se o aspirador está a espera de carregar ou de despejar o lixo, caso contrário, verifica se o aspirador tem energia. De seguida, verifica se o aspirador está no modo poupança. Caso tal aconteça, vai procurar um carregador. Se o aspirador tiver com energia e não estiver à espera de carregar ou de despejar o lixo, então verifica se o aspirador tem a capacidade de carga cheia. Se tal não acontecer, ele, para cada cor da patch tem uma função descrita mais acima neste documento Word.

4.2 Funções Implementadas para o Modelo Melhorado

StopWaitingLixoMaisSujo – Caso o aspirador esteja a aspirar o lixo mais sujo ele vai decrementando o número de ticks que o utilizador definiu.

MemorizaCarregadorNeighbours – Esta função é igual a MemorizaCarregador, mas esta é para os vizinhos que o robô perceciona no InfoNeighbours, e portanto, percorre a lista do aspirador e do aspirador vizinho e memoriza os carregadores que ainda não tenham sido memorizados pelo vizinho.

MemorizaLixo – Os aspiradores memorizam as coordenadas do depósito do lixo.

ReproduzAspiradores – Reproduz os aspiradores, caso o utilizador tenha ativado o switch.

InfoNeighbour – Melhoramos esta função para passar aos vizinhos também as coordenadas do depósito caso a tenha.

Sensores – Caso haja na vizinhança lixo, o aspirador vai para a patch com lixo e aspira-a. Caso o aspirador passe ao lado de um aspirador ou depósito vai guardar as suas coordenadas para se mais tarde precisar.

CheckFullCarregadores – Verifica se já foram todos os carregadores memorizados.

CheckIfCarregadores – Verifica se ainda há carregadores para ser memorizados

AheadLixo – Função que direciona o aspirador para o depósito que tem as coordenadas na memória. mas para os depósitos.

5. Interface

A interface é a parte que o utilizador vê, portanto tentamos deixar o mais simples e compreensível possível.

5.1 Base

A interface contém:

- **percentagemVermelho** – Percentagem de lixo existente no tabuleiro.
- **nPatchesAzul** – Quantidade de carregadores no tabuleiro.
- **nPatchesBranco** – Quantidade de obstáculos no tabuleiro. Botões "Setup" e "Go".
- **naspiradores** – Quantidade de aspiradores no tabuleiro.
- **quantEnergia** – Energia inicial dos aspiradores.
- **capCarga** – Capacidade máxima de lixo que os aspiradores conseguem transportar.
- **poupancaEnergia** – A partir de que ponto de energia, os aspiradores entram no modo poupança.
- **tempoTicksCarregamento** – Quantidade ticks o aspirador precisa para carregar 100% da bateria.
- **tempoTicksDespejarlixo** – Quantidade ticks o aspirador precisa para despejar todo o lixo que tinha acumulado.
- **Modelo** – Tela onde tudo acontece visualmente para o utilizador.

5.2 Melhorado – Implementações

- **reproducaoSwitch** – Switch para se quisermos a reprodução na nossa simulação
- **LixoMaisSujoSwitch** – Se quisermos que haja lixo mais sujo, de cor amarela
- **tempoTicksApanhaLixo** – Número de ticks que o aspirador vai demorar a aspirar o lixo mais sujo
- **LixoMaisSujo** – Quantidade de lixo mais sujo que o utilizador quer para o tabuleiro

- **nLixoMaxDespejado** – Número de quantidades de vezes que o aspirador tem de passar no depósito do lixo para ficar com os sensores avariados

6. Experiências

Aqui vamos realizar experiências para testar os modelos implementados (Base e Melhorado), fazendo um total de 10 iterações (repetições) por cada experiência. Vamos anotar o número médio de ticks, de lixo por apanhar e de aspiradores que sobreviveram.

Para estas experiências tomamos como padrão, os seguintes dados:

Modelo Base

- nPatchesBranco: 15
- quantEnergia: 100
- poupancaEnergia: 30
- capCarga: 20
- nPatchesAzul: 4
- percentagemVermelho: 10
- tempoTicksDespejarlixo: 5
- tempoTicksCarregamento: 10
- naspiradores: 20

Modelo Melhorado (juntamente com os do Modelo Base)

- LixoMaisSujo: 4
- tempoTicksApanhaLixo: 20
- nLixoMaxDespejado: 10
- LixoMaisSujoSwitch: false
- reproducaoSwitch: false

6.1 O nível de energia em que entra em poupança de bateria influencia na tarefa de limpeza do espaço?

Nesta experiência, variamos o dado de entrada poupança de energia (poupancaEnergia) entre 20 e 60, registámos as diferenças e comparamos entre modelos.

No Modelo Base com uma poupança de energia de 20, nunca conseguiram limpar o lixo todo e acabaram por ficar sem energia, e deixaram uma média de lixo por limpar de 12.4. Já com uma poupança de energia de 60, foi infinito duas vezes, ou seja, os aspiradores como tinham uma poupança muito alta acabavam por nunca conseguir chegar ao lixo pois ficavam logo sem bateria

e retornavam ao carregador e ficavam num ciclo, só não conseguiram limpar todo uma vez, deixaram uma média de lixo por apanhar de 3.2 e sobreviveram uma média de 11.9 aspiradores. Demoram muito mais com uma poupança de energia de 60 (2713.8 ticks) em comparação com uma poupança de energia de 20 (513.3 ticks), principalmente devido ao que referimos em cima.

No Modelo Melhorado com uma poupança de energia de 20, limparam sempre tudo e a média de ticks diminui drasticamente, sendo o número médio de aspiradores 10.8, ligeiramente menor que modelo base com 60 na poupança de energia. Com a poupança a 60, o número de ticks aumentou, pois, os aspiradores tiveram de ir mais vezes carregar, porém conseguiram limpar todo o lixo na mesma em relação ao número médio de aspiradores aumentou um pouco desde a nossa última experiência.

O modelo melhorado teve uma melhor resultado, algo já expectável, devido aos sensores, mas principalmente por se dirigirem para o carregador mais próximo.

6.2 O número de patches azuis influencia na tarefa de limpeza do espaço? E qual a diferença do modelo melhorado para o modelo base?

Nesta experiência, alteramos o número de carregadores presentes no tabuleiro entre 2 e 5, tanto para o modelo base como para o modelo melhorado.

No modelo base e com apenas 2 carregadores o número médio de ticks foi relativamente alto (789.8) e foi a experiência que deixaram mais lixo por apanhar com uma média de 11 patches vermelhas por limpar. Nenhuma das 10 experiências conseguiu limpar o modelo por completo pois os aspiradores ficaram sem energia. Agora acrescentamos mais 3 carregadores, fazendo um total de 5. O número de ticks desceu ligeiramente para 676.9 enquanto que a média de lixo por apanhar desceu para os 3.7. Agora com os 5 carregadores, houve uma grande diferença pois 3 das nossas 10 experiências conseguiram limpar o modelo por completo.

Passamos agora para o modelo melhorado com 2 carregadores. O número de ticks continua a descer assim como o número médio de lixo que ficou por apanhar agora nos 0.8. Com um resultado muito melhor do que o modelo base, o modelo melhorado conseguiu em 8 experiências limpar o modelo todo e ainda sobraram 10 aspiradores. Com 5 carregadores o modelo melhorado consegue ser mais vantajoso, pois diminuiu o número de ticks, consegue limpar em todas as experiências o lixo todo e ainda consegue o maior número médio de aspiradores sobreviventes, 15.7.

Ou seja, nestas nossas experiências conseguimos deixar bem vincadas as melhorias implementadas no modelo melhorado, devido a ao se dirigir sempre para o carregador mais perto.

6.3 É preferivel ter reprodução ou não na tarefa de limpeza do espaço?

Como só implementamos a reprodução no modelo melhorado as nossas experiências serão todas feitas neste mesmo modelo.

Com a reprodução a falso, os aspiradores conseguem limpar o lixo todo numa média de 334.4 ticks. Sobraram ainda, em média, 13.8 aspiradores.

Colocando o valor da reprodução a verdadeiro, as nossas experiências ficam com um resultado péssimo pois quando os aspiradores começaram a morrer vão atrapalhar a progressão dos outros colocando obstáculos no seu caminho, e por isso o resultado destas experiências não foi possível obter, ou seja, o número de ticks e o número médio de lixo foi infinito, já que a experiência nunca acaba.

Conclusão, a reprodução atrapalha muito mais do que ajuda, nas nossas experiências.

6.4 É preferivel ter o Modelo Base ou o Modelo Melhorado com lixo mais sujo?

Nesta questão tentamos pegar no modelo base simples sem modificações e compará-lo com o modelo melhorado com uma percetagem de 4% de lixo mais sujo.

O modelo base normal só conseguiu limpar 1 vez o modelo por completo com uma média de 973.2 ticks e uma média de 11.1 de lixo por apanhar.

Já o modelo melhorado sem o lixo mais sujo foi absolutamente vencedor neste duelo pois conseguiu limpar todo o modelo em todas as experiências realizadas, com ainda a sobrar 13.8 aspiradores em média por experiência. Isto tudo em 334,4, quase 1/3 dos ticks do modelo base. Com o lixo mais sujo ativo, o número de ticks aumentam, naturalmente visto que o lixo mais sujo demora um certo número de ticks a ser aspirado, mas consegue uma limpeza total do espaço e ainda com uma média de 11.8 aspiradores sobreviventes.

Como acabamos de provar, o modelo melhorado mesmo com a adição de lixo mais sujo e mais demorado de limpar consegue ser muito mais eficiente e ainda limpar tudo em menos tempo do que o modelo base.

6.5 Será que é preferivel ter mais aspiradores e não haver poupança de energia ou haver menos aspiradores e haver poupança de energia?

Quando o modo poupança é ativado, os aspiradores ignoram o lixo, passando-se a focar exclusivamente na procura de um carregador. Mas será que haver uma poupança de energia vai ajudar ou não? Usando uma energia (quantEnergia) de 65, fizemos esse teste.

No modelo base, com 70 aspiradores e o modo poupança a 0, com uma média de 69.9 ticks, os aspiradores conseguiram limpar o modelo apenas 1 vez por completo e deixando uma média de 2.1 de lixo por apanhar. Com a poupança ativa no valor de 30, o número de ticks aumenta drasticamente para 2269.3 e ainda piorou o resultado final, com nenhuma das experiências a conseguir limpar o lixo todo.

No modelo melhorado, as coisas mudam, pois, sem o modo poupança os aspiradores com uma média de apenas 71.2 ticks limpam o modelo em todas as nossas 10 experiências, sobrando ainda 41.5 aspiradores, de média. Com o modo poupança ativo, os aspiradores conseguem ser bem-sucedidos em 3 ocasiões, porém demoram uma média de 4083.2 ticks e há uma média de sobrevivência de 15 aspiradores por simulação.

Aqui conseguimos, novamente, perceber as melhorias do modelo melhorado tendo mais eficácia em relação ao Modelo Base, mas além disso, conseguimos ver que é preferível haver mais aspiradores do que haver carregadores mais "inteligentes", mesmo assim, acreditamos que com uma quantidade de energia inicial menor, o facto de haver uma poupança de energia ia prevalecer e ser muito mais vantajoso.

6.6 Será que é preferivel ter uma quantidade de energia maior mas uma capacidade máxima menor ou o reverso?

Aqui, queremos testar se é preferível ter uma quantidade de energia maior ou uma capacidade máxima maior. Pois quanto maior for a quantidade de energia, mais ticks vai limpar antes de precisar de carregar, enquanto que quanto maior for a capacidade máxima mais lixo consegue armazenar e por isso continuar a aspirar.

Com o modelo base, energia com um valor de 75 e uma capacidade de carga de apenas 10, os aspiradores têm um mau desempenho apenas limpando totalmente o modelo apenas 1 vez, com uma média de 1239.8 ticks. Com a energia menor, a 50, e a capacidade de carga máxima o desempenho piora e nenhuma das experiências consegue limpar o modelo e ainda fica com a pior média de lixo por aspirar do nosso trabalho.

No modelo melhorado, as experiências têm a tendência a melhorar e foi mesmo o que aconteceu. Com mais energia e menos carga, os aspiradores conseguiram em 6 ocasiões limpar todo o modelo, mas com 1 das experiências a dar infinito. Com menos energia e mais capacidade de armazenamento de lixo, os ticks aumentam absurdamente, a chegar à média de 4739.5 ticks por simulação, porém conseguindo limpar tudo 3 vezes. Ainda registamos 4 infinitos.

Como podemos observar, o modelo melhorado obtém, mais uma vez, melhores resultados, mas o que chama mais à atenção é o facto de ser mais produtivo usar mais energia e menos capacidade de carga do que o contrário pois mesmo que armazenem menos lixo, não precisam de ir carregar tão depressa como o contrário.

7. Conclusão

Depois de concluirmos este trabalho, sentimos que foi uma mais-valia na nossa aprendizagem e no nosso percurso académico. Para além disso, encaramos a nossa vida profissional melhor preparados para os desafios que nos esperam de futuro.