

Relatório IA: 1a parte do Projecto - IST @ 2018/2019 Grupo 47

Nesta parte do projecto tivemos de implementar uma solução para a resolução do jogo *Solitaire*.

Passámos a todos os testes à excepção dos testes 30, 31 e 32, provavelmente devido ao facto de não termos conseguido implementar na totalidade as heurísticas que escolhemos e que nos pareceram apropriadas no contexto deste problema.

Escolhemos a soma de 2 heurísticas para a solução do problema :

➤ O número de ilhas

Em que uma ilha corresponde a um berlinde ou a um conjunto de berlindes que não têm moves possíveis, isto é que estão rodeados de posições desocupadas (“_”) ou posições bloqueadas (“X”).

Uma vez que para executar uma jogada é necessário extrair uma peça da board, vizinha da posição do berlinde a jogar, então quanto maior for o número de ilhas mais difícil será a resolução do problema.

➤ A distância entre ilhas

Considerando que quanto menor for essa distância, mais próximas estão as ilhas, logo mais difícil seria a resolução do problema.

Apesar do raciocínio explicado em cima, conseguimos apenas aplicar uma heurística semelhante à 1ª, que calcula se os “cantos” dos boards estão ocupados com berlindes, analogamente se os cantos são ilhas.

Vamos analisar e comparar em cada um dos seguintes tabuleiros, em termos do tempo de execução, número de nós expandidos e número de nós gerados, obtidos por procura em profundidade.

Tabuleiros:

B1 = [["_", "O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O", "O"]]

B2 = [["_O", "O", "O", "X"], ["O", "O", "O", "O"], ["O", "O", "O", "O"], ["O", "O", "O", "O"], ["O", "O", "O", "O"], ["O", "O", "O", "O"]]

B3 = [["_O", "O", "O", "X", "X"], ["O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O"], ["O", "O", "O", "O", "O"]]

| | Tipo de Procura | Tempo (seg) | Nós Expandidos | Nós Gerados |
|------------------|-----------------|-------------|----------------|-------------|
| Tabuleiro B1 5x5 | DFS | 0.001 | 14 | 34 |
| Tabuleiro B2 4x4 | DFS | 0.327 | 5987 | 11990 |
| Tabuleiro B3 5x4 | DFS | 3.777 | 53637 | 107301 |

Em todas as procuras, mantendo as dimensões e aumentando o número de peças ou aumentando as dimensões e mantendo o número de peças, o tempo aumenta pelo facto de haver mais nós a verificar e a expandir, pois há mais opções de jogo.

E em todos os casos a procura em profundidade é a pior, pois a solução encontra-se na ponta oposta à do início da procura.

Como o custo de caminho é $c+1$, a procura A^* acaba por não diferir muito de uma procura greedy em termos de números de nós gerados e expandidos. Assim em termos de eficiência, teoricamente, a DFS seria a pior e a procura greedy e o A^* estão muito perto uma da outra.