UNIVERSIDADE DE ÉVORA

Relatório Inteligência Artificial

Rúben Peixoto e Vanessa Santos
37514 e 34191

Maio 2020

1 Introdução

Este projecto usa predicados que, possivelmente, só funcionam em SWI Prolog.

Neste trabalho foram testados dois algoritmos: o hill climbing e o genetic algorithm. Como o hill climbing entra em loop em minimos locais, o grupo decidiu utilizar outro algoritmo. No entanto este também parece entrar em loop.

2 Hill Climbing

Para usar este algoritmo, basta remover os comentários que vão da linha 216 até à linha 233 e usar o predicado "start_hill.".

Este algoritmo funciona, só que fica em loop em mínimos locais. Por exemplo, foi testado num tabuleiro 4x4 e uma vezes chegava a um estado final, outras fica preso num estado intermédio.

Por isso pode-se afirmar que este algoritmo, se não fica-se preso em mínimos locais, colocaria 20 rainhas num tabuleiro 20x20 sem nenhum problema.

3 Genetic Algorithm

Para tentar resolver o problema do loop no hill climbing, o grupo decidiu usar o algoritmo "genetic algorithm". Isto porque, este algoritmo usa estados aleatórios para criar o estado seguinte.

Embora este algoritmo também entre em loop como o hill climbing, há certos tamanhos do tabuleiro em que este apresenta sempre um resultado final. Esses tamanhos são 4x4, 5x5, 6x6, 7x7, 8x8 e 9x9.

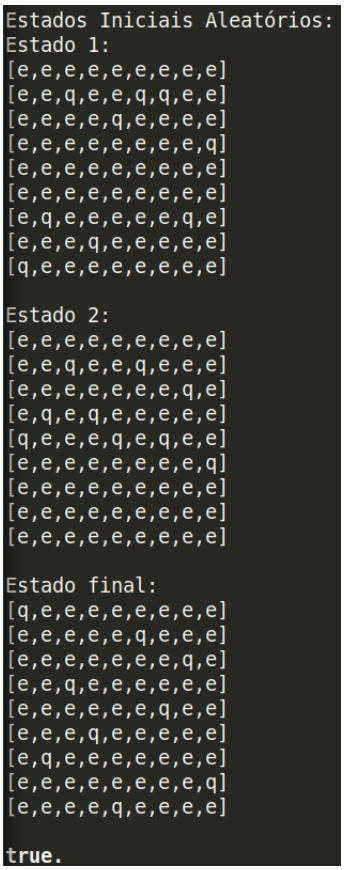

Figure 1: Tabuleiro 9x9

A partir do tabuleiro 10x10, o algoritmo já entra em loop. Para resolver este problema o grupo tentou inserir estados aleatórios.

Com isto o grupo, fica pensa que este loop não se deve ao facto de o programa estar preso a um estado mas sim ao facto de que pode existir estados aleatórios que não ajudam o algoritmo a encontrar um solução.

Por isso, tal como o hill climbing, este algoritmo se não entrasse em loop, conseguiria colocar 20 rainhas num tabuleiro de 20x20.

Este algoritmo começa por adicionar 2 estados aleatórios a uma lista. O algoritmo genetic algorithm vai buscar, 2 estados em posições aleatórias e junta características diferentes dos 2 estados num estado filho. Posto isto coloca-o na lista. Depois disto o algoritmo, em ciclo, volta a aceder à lista com os estados e volta a fazer o mesmo.

Para executar este algoritmo é necessário executar o predicado "start.".

4 Bibliografia

- Slides do Professor
- Artificial Intelligence, A Modern Approach.