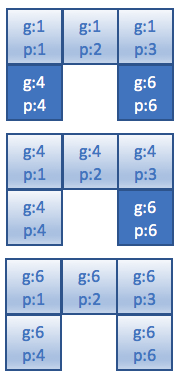
**1º Projeto IA – Relatório 2017 – 2018**

Grupo tg028 – Diogo Vilela 84710 e Diogo Redin 84711

Análise Teórica

Para a representação de um tabuleiro é utilizada uma lista de listas em que cada célula contém a cor da peça. Cada peça tem um ID específico e um ID de conjunto. Inicialmente todas as peças são consideradas um conjunto individual.

A heurística implementada para a procura informada consiste no cálculo do número de conjuntos de peças presentes num tabuleiro. Este cálculo não é otimista, portanto a heurística implementada não é admissível.

Para encontrar conjuntos de peças no tabuleiro, este é percorrido linha a linha, coluna a coluna e quando são encontradas peças adjacentes com a mesma cor, o ID do conjunto da peça é propagado para os adjacentes. A complexidade temporal deste procedimento é de , sendo que na maior parte dos casos é .

Fig. 1 – Propagação de IDs.

Resultados

Apresentamos de seguida os testes apresentados no enunciado e aplicados ao nosso programa. Os testes foram corridos num computador com um processador i5 de 2.3GHz. Todos os tempos obtidos são a média de cinco repetições do respetivo teste.

**Teste 1: Tabuleiro de 4x5 (linhas x colunas) com 2 cores sem solução**

*Input: [[1,2,1,2,1], [2,1,2,1,2], [1,2,1,2,1], [2,1,2,1,2]]*

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Procura em Profundidade Primeiro** | **Procura Gananciosa** | **A\*** |
| *Tempo de execução (s)* | 0.0726s | 0.0744s | 0.0766s |
| *Nós expandidos* | 1 | 1 | 1 |
| *Nós gerados* | 0 | 0 | 0 |

**Teste 2: Tabuleiro de 4x5 (linhas x colunas) com 3 cores**

*Input: [[1,2,2,3,3], [2,2,2,1,3], [1,2,2,2,2], [1,1,1,1,1]]*

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Procura em Profundidade Primeiro** | **Procura Gananciosa** | **A\*** |
| *Tempo de execução (s)* | 0.0754s | 0.0780s | 0.0766s |
| *Nós expandidos* | 4 | 3 | 4 |
| *Nós gerados* | 7 | 6 | 7 |

**Teste 3: Tabuleiro de 10x4 (linhas x colunas) com 3 cores sem solução**

*Input: [[3,1,3,2], [1,1,1,3], [1,3,2,1], [1,1,3,3], [3,3,1,2], [2,2,2,2], [3,1,2,3], [2,3,2,3], [5,1,1,3], [4,5,1,2]]*

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Procura em Profundidade Primeiro** | **Procura Gananciosa** | **A\*** |
| *Tempo de execução (s)* | 12.4554s | 25.9684s | 26.6036s |
| *Nós expandidos* | 74702 | 74702 | 74702 |
| *Nós gerados* | 74701 | 74701 | 74701 |

**Teste 4: Tabuleiro de 10x4 (linhas x colunas) com 3 cores**

*Input: [[3,1,3,2], [1,1,1,3], [1,3,2,1], [1,1,3,3], [3,3,1,2], [2,2,2,2], [3,1,2,3], [2,3,2,3], [2,1,1,3], [2,3,1,2]]*

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Procura em Profundidade Primeiro** | **Procura Gananciosa** | **A\*** |
| *Tempo de execução (s)* | 0.0916s | 0.0970s | 0.0880s |
| *Nós expandidos* | 54 | 42 | 24 |
| *Nós gerados* | 85 | 59 | 43 |

**Teste 5: Tabuleiro de 10x4 (linhas x colunas) com 5 cores**

*Input: [[1,1,5,3], [5,3,5,3], [1,2,5,4], [5,2,1,4], [5,3,5,1], [5,3,4,4], [5,5,2,5], [1,1,3,1], [1,2,1,3], [3,3,5,5]]*

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Procura em Profundidade Primeiro** | **Procura Gananciosa** | **A\*** |
| *Tempo de execução (s)* | 8m12s | 0.1924s | 0.0968s |
| *Nós expandidos* | 3123308 | 256 | 16 |
| *Nós gerados* | 3123363 | 319 | 91 |

Análise de Resultados

**Completude:** uma vez que cada ação no jogo é irreversível, não é possível repetir sequências de estados no tabuleiro e, portanto, se existir uma solução esta é sempre encontrada por todos os métodos.

**Eficiência e Comparação:** teoricamente, a procura em profundidade primeiro tem complexidade , a procura gananciosa tem complexidade e a procura A\* tem complexidade . *B - Fator de ramificação, r - fator de expansão, m - profundidade.*

Comparando os resultados dos testes é possível verificar que estas complexidades se verificam. Nomeadamente, quando é aumentado o fator de ramificação, neste caso, o número de cores, o número de nós expandidos e gerados aumenta significativamente sendo que as procuras informadas mantêm a mesma eficiência.