

# Relatório do Projeto 1 de Inteligência Artificial

Mihail Brinza

83533

Ricardo Brancas

83557

20 de Outubro de 2017

## 1 Resultados Exeprimentais

### 1.1 Breve descrição dos problemas utilizados

**Problema 1** [4x5], 2 cores, sem solução

**Problema 2** [4x5], 3 cores

**Problema 3** [10x4], 5 cores, sem solução

**Problema 4** [10x4], 3 cores

**Problema 5** [10x4], 5 cores

### 1.2 Resultados

Problema	DFS			Greedy			A*		
	t	g	e	t	g	e	t	g	e
1	0.0003	0	1	0.0005	0	1	0.0006	0	1
2	0.0006	7	4	0.0009	6	3	0.0009	7	4
3	12.574	74701	74702	17.673	74701	74702	12.833	74701	74702
4	0.0057	85	54	0.0144	59	42	0.0119	43	24
5	208.09	3123363	3123308	0.0539	220	149	0.0306	91	16

**t** refere-se ao tempo demorado; **g** ao número de nós gerados e **e** ao número de nós expandidos.

Tabela 1: Resultados da execução dos vários problemas.

## 2 Análise

### Caracterização do Espaço de Estados

Tomando um qualquer tabuleiro  $b$  e sabendo que uma ação  $a$  consiste na remoção de um grupo de cardinalidade  $\geq 2$ , conclui-se que o tabuleiro resultante de aplicar  $a$  a  $b$  tem estritamente menos peças que o tabuleiro original. Como tal, conclui-se que não podem existir ciclos no espaço de estados, pois nunca é possível o tabuleiro voltar a ganhar peças.

Em acréscimo, uma vez que o número de peças é finito, o comprimento máximo de um caminho no espaço de estados é também finito (correspondendo à situação de se removerem todas as peças do tabuleiro, em grupos apenas de 2). Como tal não podem existir ramos infinitos no espaço de estados.

Em último lugar, temos que para cada estado existe um número máximo de sucessores diretos finito, uma vez que cada ação corresponde a remover um grupo e o número de grupos é finito. Consequentemente o *branching factor*,  $b^*$ , é também finito.

**Completude** No problema Same Game, qualquer uma das pesquisas estudadas é completa uma vez que o espaço de estados é acíclico e tanto o comprimento máximo de um caminho como o *branching factor* são finitos.

**Eficiência** Dos 3 métodos de pesquisa abordados, os mais eficientes (em termos de tempo/nós gerados) são claramente as procuras informadas, neste caso a Procura Gananciosa e a Procura A\*. Isto deve-se ao facto das procuras informadas usarem heurísticas para direcionar a pesquisa em direção ao estado objetivo, reduzindo efetivamente o *branching factor*. Por este motivo, a discrepância entre as procuras informadas e não informadas é mais notória quando existe um grande número de grupos alternativos que podem ser removidos ( $>$  *branching factor*).

Através da análise de várias instâncias do problema, concluímos ainda, que um dos fatores determinantes para a complexidade de uma dada instância é a relação entre o número de cores e o tamanho do tabuleiro, tal como no clássico jogo *minesweeper* (neste caso número de bombas vs. tamanho do tabuleiro); pois num tabuleiro muito grande com poucas cores a solução é normalmente mais simples (uma vez que os grupos são em geral maiores). Em contraste, num tabuleiro pequeno com muitas cores os grupos são em geral muito pequenos o que dificulta a resolução do problema.

**Heurística** Na implementação do nosso problema Same Game usámos a seguinte função heurística:

$$h(n) = \sum_{g \in groups(n)} \begin{cases} 1, & size(g) \geq 2 \\ 3, & size(g) = 1 \end{cases}$$

Para cada grupo de cardinalidade  $\geq 2$  somamos 1 ao valor de  $h(n)$  uma vez que, no melhor caso, esse grupo irá ser removido apenas numa ação; para cada grupo de cardinalidade 1, somamos 3 de forma a dirigir a nossa pesquisa de modo a evitar grupos de cardinalidade 1, uma vez que esses grupos são provavelmente mais difíceis de remover que grupos com maiores cardinalidades. Experimentalmente constatámos que esta heurística produz ligeiramente menos nós do que não fazendo diferenciação na cardinalidade dos grupos.

Esta heurística é claramente não admissível, uma vez que remover um grupo pode implicar uma alteração radical no número de grupos restantes (pois vários grupos separados podem juntar-se num só). No entanto, como este não é um problema de optimalidade e sim de satisfação, não é importante que a heurística seja admissível mas sim que conduza o mais rapidamente possível ao estado objetivo.