



Universidade do Porto  
Faculdade de Engenharia

**FEUP**

## **A5: Pesquisa aplicada à resolução do jogo**

### **Pukoban**

*Relatório Intercalar*

Inteligência Artificial

3º ano do Mestrado Integrado em Engenharia Informática e Computação

Elementos do Grupo:

Bruno Miguel Faustino Moreno – up201504781 – up201504781@fe.up.pt

Francisco Teixeira Lopes – ei11056 – ei11056@fe.up.pt

8 de Abril de 2018

## Objectivo

O objectivo do trabalho incide sobre a resolução do jogo Pukoban, usando métodos de pesquisa. Para esse efeito será usado A\* inicialmente, e posteriormente, métodos como pesquisa em largura e aprofundamento iterativo para comparação de desempenho.

## Descrição

### Especificação

O jogo Pukoban desenrola-se num mapa com objectos e movimentos em grelha. O objectivo é deslocar um conjunto de caixas de forma a ficarem sobre os campos marcados como destino, para esta deslocação, é possível tanto empurrar como puxar as caixas.

Como o jogo consiste num espaço em grelha, cada estado é representado pela posição dos vários elementos dinâmicos. Sendo que, os elementos estáticos, como paredes e formato do nível, não são guardados directamente nos estados mas entram na geração dos estados possíveis.

Sendo o principal algoritmo a implementar o A\*, a função de transição será:

$$f^*(n) = g(n) + h^*(n)$$

Onde  $g(n)$  é o custo até ao estado actual e  $h^*(n)$  é o custo estimado para chegar à solução a partir do estado actual. A heurística a implementar baseia-se na distância de cada objectivo à caixa mais próxima, na verdade, é o somatório das distâncias de cada objectivo à caixa mais próxima. A distância é calculada como sendo a distância na grelha actual, contando com obstáculos, e não apenas como sendo uma distância em linha recta. Além disso, é ainda considerado no cálculo da distância, o número de vezes que a caixa tem de mudar de direcção em 90 graus, pois isto implica o jogador ter de fazer no mínimo 2 jogadas.

$$\sum_{i=0}^n d(objectivo_i \text{ à caixa mais próxima}) + 2 * nCurvasNecessárias$$



*Figura 1 - exemplo estado de jogo com arte placeholder*

A figura acima exemplifica a heurística para um objectivo específico, a rota a azul teria um peso de 3 já que a caixa não tem de mudar de direcção, por outro lado, a rota a preto teria um peso de 2 na distância, mas acrescia um peso de 2 pela necessidade de mudar uma vez de direcção, para um total de 4. A caixa mais afastada seria inicialmente favorecida devido a isto (peso de 3 vs peso de 4). Na verdade, estando o jogador mais perto da caixa com um peso superior, faria sentido para um jogador humano começar por essa caixa, contudo, para evitar o risco de sobreestimativa da heurística, não é considerada a posição do jogador em relação às caixas. Neste caso, seria a caixa mais próxima a jogada ideal mas existe uma infinidade de cenários em que a caixa mais próxima do jogador e do objectivo não é a jogada correta.

O algoritmo de pesquisa a utilizar será o A\* com a função de transição e heurística enunciada acima. Os estados serão gerados baseando-se nas jogadas possíveis a partir do nó atual, verificando nós duplicados e considerando-os inválidos. Para um nível muito simples de Pukoban, o algoritmo comportaria-se da seguinte forma:

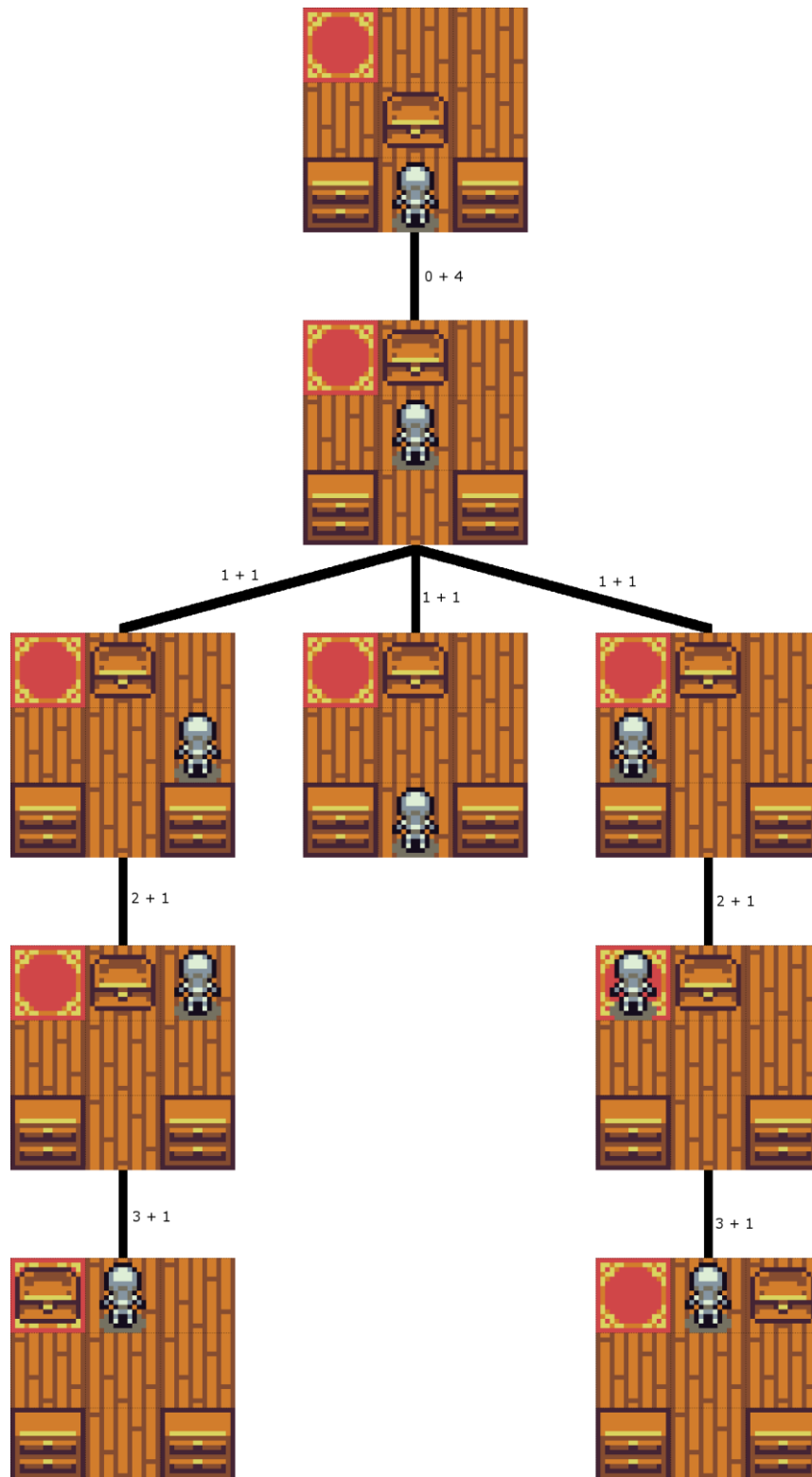


Figura 2 - aplicação do algoritmo

Imaginando que o algoritmo decidia prosseguir primeiro pelo ramo da direita, visto ambos os ramos terem o mesmo custo, chegaria a um estado em que a próxima jogada teria um custo heurístico de 2, pois a caixa afastou-se do objectivo, enquanto que no ramo esquerdo continuaria a 1, levando à eventual solução do nível. A heurística acaba por favorecer estados que aproximem as caixas do objectivo, sendo que, só expande os outros estados se estes primeiros não levarem a uma solução.

### Trabalho Efectuado

Até à data de entrega do relatório apenas foram decididas a linguagem e frameworks a usar para o trabalho, a qual será Java com a framework libGDX e a API de grafos JGraphT. Além disso, foi feito o planeamento teórico da heurística e metodologia a utilizar para a resolução do problema.

### Resultados esperados e forma de avaliação

Para avaliar os resultados, criarão-se níveis com melhor solução conhecida para poder comparar ao resultado obtido pela implementação do algoritmo A\*. Além disso, irá-se ainda comparar outros algoritmos aos resultados obtidos pelo A\* em relação à solução ideal. Espera-se que o A\* encontre sempre a solução ideal se bem que nem sempre no tempo ideal, enquanto que, nos outros algoritmos espera-se um tempo de execução bastante superior, ou então, uma solução não ideal (ex. aprofundamento iterativo).

### Conclusões

Após escrita do relatório e planeamento do método de resolução do problema, o grupo tem uma melhor apreciação pelo trabalho necessário para conduzir ao eventual produto final. Embora a implementação do algoritmo e da heurística escolhida não seja particularmente difícil, a parte mais trabalhosa e possivelmente mais interessante, trata-se da comparação entre os resultados de diferentes algoritmos de pesquisa. O grupo encontra-se curioso quanto aos resultados esperados descritos na seção “Resultados esperados e forma de avaliação”, e se, verdadeiramente serão os resultados observados. Não tendo outras conclusões a adicionar, termina-se este relatório intercalar quanto à resolução do jogo Pukoban.