

Trabalho Prático I – Busca no Pacman*

Diego Ascânio Santos¹

¹Departamento de Computação – Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG)

Av. Amazonas, 7675 - Nova Gameleira, Belo Horizonte - MG, 30510-000 – Brazil

dascaniosantos@gmail.com

Abstract. *At this schoolwork, the Pacman agent needs to find paths in the maze, to reach a goal or to pick food efficiently. The schoolwork's goal is to program search algorithms to apply them at Pacman's scenario.*

Resumo. *Neste trabalho, o agente Pacman tem que encontrar caminhos no labirinto, tanto para chegar a um destino quanto para coletar comida eficientemente. O objetivo do trabalho será programar algoritmos de busca e aplicá-los ao cenário do Pacman.*

1. Introdução

O Pacman é um personagem e jogo desenvolvido pela Namco lançado em 1980 que consiste em um personagem amarelo explorando um labirinto em busca de comida (pontos brancos na tela) e lutando pela sobrevivência, fugindo de fantasmas existentes no mapa. O jogo é tido como um dos mais populares de todos os tempos na história dos videogames e por isso, a sua importância ultrapassou os limites dos consoles daquele tempo e persiste até os dias atuais, onde o jogo virou tema para desenhos animados, brinquedos, bugigangas, e até trabalhos de inteligência artificial! O jogo foi utilizado no desenvolvimento de um projeto didático para o curso introdutório de inteligência artificial da universidade de Berkley, com a finalidade de ensinar os conceitos básicos da disciplina. Baseado nesse projeto, escrito em linguagem Python, encontram-se os códigos dos algoritmos de busca desenvolvidos neste trabalho para solucionar o problema de encontrar comida(s) nos labirintos do jogo.

2. Algoritmos de Busca

Um algoritmo de busca é um “Algoritmo que toma um problema como entrada e retorna a solução para o problema, geralmente após resolver um número possível de soluções.”(1). Essa definição descreve perfeitamente os tipos de algoritmo desenvolvidos para resolver o problema deste trabalho, que consiste em: Dado um labirinto do jogo Pacman, contendo um ou mais pratos de comida, encontrar o (os) caminho (s) que levem o nosso gracioso personagem de encontro a ele(s).

* Adaptado para OpenOffice.org 1.1 feita por Roland Teodorowitsch (roland@ulbra.tche.br) em 29 mar. 2005.

O labirinto do jogo deve ser mapeado para uma estrutura de dados grafo, pois os algoritmos de busca desenvolvidos neste trabalho (DFS, BFS, UCS e A*) são feitos para trabalhar com esta estrutura.

3. Desenvolvimento do trabalho

O trabalho foi desenvolvido na linguagem Python utilizando o editor de textos vim para a construção dos códigos. Os códigos foram desenvolvidos a partir dos exemplos vistos em sala de aula e de pseudocódigos do livro de algoritmos do Cormen – Algoritmos: Teoria e Prática 2ª Edição – procurando mantê-los o mais parecido possível com seu pseudocódigo, para facilitar a legibilidade e manter a simplicidade do algoritmo, com isso seguindo as boas práticas e recomendações da linguagem Python.

4. Execução do trabalho

A execução do trabalho ocorreu sempre pela linha de comando, executando-se os passos listados no roteiro do trabalho para a execução de cada uma das soluções(2)

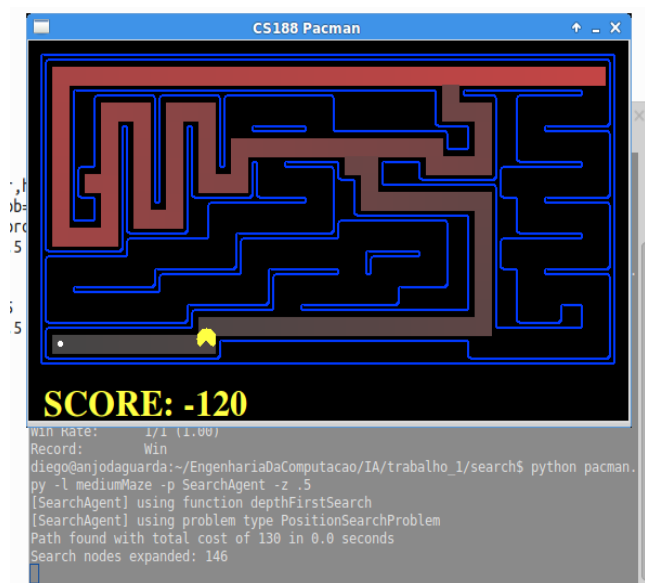


Figura 1. Exemplo de execução do Pacman

5. Respostas obtidas

No roteiro do trabalho são propostas cinco perguntas a serem respondidas pela interpretação do comportamento do pacman em relação aos algoritmos (DFS, BFS, UCS e A*) implementados. Eis as perguntas e respostas:

(Pergunta 1) A ordem de exploração foi de acordo com o esperado? O Pacman realmente passa por todos os estados explorados no seu caminho para o objetivo?

(Resposta) Sim, pois o pacman no medium maze pela implementação do DFS irá visitar os nós de acordo com a ordem da pilha de exploração e na pilha, o dfs adiciona o caminho mais a esquerda por último (na primeira bifurcação que existe no mapa) e continua explorando esse caminho, adicionando novos nós e retirando-os da pilha a medida em que seus adjacentes estão nos estados visitados. Por último, na sala de 4x4 do labirinto, ele toma o caminho ao sul que o levará ao objetivo. O pacman também não passa por todos os estados explorados do objetivo, mas isso ocorre por conta da implementação do mapa de arestas, que remove loops e descarta eventuais becos sem saída, ao invés de obrigar o pacman a retornar pelo caminho de onde veio.

(Pergunta 2) Essa é uma solução ótima? Senão, o que a busca em profundidade está fazendo de errado?

(Resposta) Não, pois como dito anteriormente, existe um caminho de menor custo e esse caminho que leva ao objetivo de forma ótima não é explorado, pois o objetivo é encontrado antes.

(Pergunta 3) A busca BFS encontra a solução ótima? Senão, verifique a sua implementação. Se o código foi escrito de maneira correta, ele deve funcionar também para o quebra cabeças de 8 peças (seção 3.2 do livro-texto) sem modificações.

(Resposta) Sim, pois ao testar esse maze com o A*, o custo foi o mesmo encontrado em BFS e o A* retorna soluções ótimas, pois a heurística (Manhattan Heuristic) é admissível.

(Pergunta 4) O que acontece em openMaze para várias estratégias de busca?

(Resposta) Para a solução DFS, 819 nós são explorados (o maior número possível), o que comprova que o DFS não é bom para caminhos abertos, uma vez que ao visitar um estado ele já exclui o mesmo do caminho para não entrar em loop e segue em frente e faz um caminho esquerda, direita, direita, esquerda, ao invés de descer direto pelo caminho que pode ser atingido verticalmente, tendo um custo muito grande. O BFS expandiu um número de nós um pouco menor e encontrou um caminho mais eficiente para o objetivo – o mesmo encontrado na busca de custo uniforme e no A*, que além de encontrarem esse caminho, encontraram com expansões menores ao BFS e o A* teve uma eficiência maior do que os demais.

6. Referências

1. Burlamaqui, Aquiles – “Aula 10 Algoritmos de Busca”, disponível em:
<http://aquilesburlamaqui.wdfiles.com/local--files/2008-1-uern-novacruz-ca/Aula10.ppt>
2. MARTINS, F. V. C - “Trabalho Prático”, disponível em:
<https://www.dropbox.com/s/yg7eup45c4pv3lj/Trabalho%201.pdf?dl=0>