Trabalho 1

April 14, 2019

1 Comparando algorítmos de busca informada e não informada

João Vitor Araki Gonçalves (176353)

1.1 Introdução

O objetivo desse trabalho é comparar a performance de diferentes algorítmos de busca no problema de um robô tentando se navegar num lambirinto.

O desenvolvimento do trabalho foi baseado nas implementações de algorítmos de busca do pacote aima-python, foram utilizadas as abordagens **breadth first search**, **depth first search** e **a-star** com as heurísticas de distância manhattan e distância euclidiana.

Todos esses algorítmos forma comparados num labirinto de 60 por 60 gerado randomicamente com paredes intransponíveis, todos foram testados no mesmo labirinto para garantir comparabilidade.

O objetivo desse trabalho é comparar como diferentes algorítmos de busca se comportam em certos cenários e como eles se comparam entre si. Para isso foi medido o número de estados visitados por cada um, a otimicidade da solução final e o tempo levado na execução do mesmo.

1.2 O estado e ações

Inicialmente o estado foi modelado como (x, y, angle), onde **x e y** é o ponto onde o robô está representado no labirinto, e o *angle* é a direção em que o robô está virado, porém isso causou alguns problemas devido a forma que o **aima** guarda estados passados, eles são todos guardados em um **set** e comparados por inteiro, então é possível que o robô retorne para um mesmo ponto no mapa, desde que esteja virado para outra direção. Isso tornou os resultados confusos e difíceis de se comparar, principalmente no caso do dfs, então foi decidido simplificar a representação do problema.

O estado é estruturado como (x, y) onde **x e y** compõe o ponto onde o robô se encontra no mapa, e ele possui ações de **GO_FOWARD**, **GO_LEFT**, **GO_RIGHT**, **GO_UP** e **GO_DOWN**. Portanto o estado da direção que ele está virado foi removida. Esse mapeamento simplificado do estado considera que o robô não tem custo para ir em outra direção, ou seja, que ele consegue ir para os quatro sentidos sem precisar se virar, como esse exemplo

Essa abordagem é equivalente à se ter o parâmetro de angulo, mas com ele ser limitado aos valores 0, 90, 180 e 270

1.3 Os algorítmos

Nessa seção vamos fazer uma análise dos diferentes algorítmos utilizados.