

Trabalho de IA

Nesse relatório, vou explicar brevemente os conceitos dos algoritmos de busca: largura, profundidade, uniforme, A*; e seus resultados

FUNDAMENTAÇÃO TEÓRICA:

Largura

Seu funcionamento consiste em visitar os nós em camadas, depois de visitar toda a camada ele expande os nós a serem visitados para mais uma profundidade, seguindo esse fluxo até encontrar o objetivo, neste cenário de problema ele é visto como um algoritmo de força bruta devido ao posicionamento do nó início e o nó objetivo final, que estão em completo polos opostos.

Profundidade

Ao contrário do de largura ele visita os nós indo primeiramente nos filhos, caso o caminho dos filhos não leve ao destino ele retorna ao último pai que havia filhos para se seguir e vai fazendo essa consulta por profundidade até encontrar o objetivo, ele não leva em consideração o caminho que está percorrendo apenas o primeiro nó em que ele recebe e que não foi expandido, repetindo até chegar no nó objetivo.

A*

Utiliza-se de uma métrica para navegar no grafo, essa métrica leva em consideração a distância do início + a distância do objetivo ($F_Cost(n) = G(n) + H(n)$), a cada nó expandido é visto os nós em aberto e é escolhido o que possui a menor métrica (F_Cost) para ser expandido novamente, até alcançar o objetivo final.

Uniforme

Semelhante ao A* porém a sua métrica não leva em consideração a distância do nó que está em relação ao objetivo, apenas a distância do nó de início.

EXPERIMENTOS:

Configuração do computador: Intel Core i3 6100U , 8GB DDR4 2400mhz , Intel HD Graphics 520

Para cada experimento será realizado 10x com a mesma seed para poder extrair a média do tempo de processamento, pois em alguns casos o tempo dobrava ou era a metade, para evitar discrepância realizei a média do tempo.

Considerando-se o funcionamento dos algoritmos e em um cenário como um labirinto de 300x300 eu estimo que:

Profundidade:

O algoritmo mais veloz será o de profundidade pois o cenário tem um escopo limitado à matriz e o objetivo fica no lado oposto do alvo, permitindo que uma busca direta de profundidade seja rápida o suficiente para ultrapassar os demais, porém será o mais custoso em caminho e em Passos.

Largura:

O algoritmo mais lento eu presumia que seria o de largura (o resultado desmente minha presunção) pois ele consulta praticamente todas as células do labirinto (devido a posição do alvo) para identificar o caminho, com a exploração de largura e considerando o problema atual ele conseguiu varrer praticamente todo o mapa, conseguindo definir o menor caminho possível para o objetivo, ele acabou se tornando o algoritmo de força bruta.

A*:

O algoritmo mais versátil é o A*, pela sua métrica de prioridade que usa, torna-o não o mais rápido nem com menor custo. Ele tem um tempo de processamento relativamente rápido e encontra um caminho também relativamente próximo do menor neste cenário.

Uniform

Esse algoritmo achava que seria tão veloz quanto o A* por sua semelhança, diferenciando-se apenas no critério que é pela distância do estado com o nó inicial em vez de considerar início e fim igual o A*.

RESULTADOS:

Tipo:	Path Cost:	Steps:	Time:
breadth	436.81	320	14.4396
A*	466.87	360	2.6237
Depth	735.65	635	0.2405
Uniform	437.64	320	21.8628

Tempo estimado pela média de 10 repetições

Os esperados foram: Largura, A* e Profundidade.

O de custo Uniforme, por ter semelhança ao A* Acreditava que seria um competidor justo, ao testar notei que ele foi o mais lento de todos e não encontrou o menor caminho igual o de busca de largura. O motivo dele ser tão ruim acredito que seja por conta da complexidade que ele possui, que é semelhante ao do A*, porém o seu critério não leva em consideração a distância do objetivo, apenas a distância do ponto de início, tornando ele ineficiente no aspecto de velocidade. A vantagem é que ele encontra um caminho curto, não o melhor, e demora mais tempo do que o de largura, que praticamente está funcionando neste problema como força bruta por conta do início e fim estarem em lados completamente opostos.

O algoritmo de busca de largura se comportando como um de força bruta atingiu o esperado em demorar bastante tempo e encontrar o melhor resultado.

O algoritmo de A* por conta dos critérios que considera ponto de início e fim realizou um bom trabalho em achar um caminho rápido em um custo de pouquíssimo tempo comparado aos demais.

O algoritmo de profundidade é o mais banal, ele é corre em uma linha até achar o objetivo, se não tem muitos obstáculos ele consegue chegar rapidamente, seu critério é muito simples, o primeiro que ele ver expande, facilitando bastante neste cenário a identificação do objetivo, tornando-o o mais rápido de todos porém o que possui o pior peso e caminho de todos devido a sua inteligência.