Alunos

Douglas da Silva Pereira Veras | Matrícula: 20190113674

José Jesus do Monte Andrade de Morais | Matrícula: 20190100030

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Introdução

O presente relatório descreve a implementação e aplicação do algoritmo A\* para busca de rotas na Romênia. O algoritmo utiliza a Busca em Largura como modelo, fazendo uso de uma fila de prioridade para a fronteira de exploração. O objetivo é encontrar o caminho de menor custo entre duas cidades, levando em consideração tanto o custo acumulado como uma heurística estimada.

Implementação

A implementação do algoritmo A\* é composta por várias classes e funções. Inicialmente, temos a classe MinHeap, que representa a fila de prioridade e é responsável por manter os nós ordenados pelo valor da função f(n). A classe utiliza um heap mínimo e implementa métodos como adiciona e remove\_min para adicionar e remover elementos da fila de acordo com a prioridade.

Em seguida, temos a classe Node, que representa um nó no grafo. Cada nó possui um estado, um pai (nó anterior), uma ação, um custo acumulado (g) e uma heurística estimada (h). A função f(n) é definida como a soma de g e h.

A classe PriorityQueue é responsável por encapsular a fila de prioridade (MinHeap) e oferece métodos simples para adicionar e remover elementos.

O algoritmo principal é a função a\_star, que recebe o grafo, a cidade de origem, a cidade de destino e a função heurística como parâmetros. Essa função realiza a busca do caminho de menor custo utilizando o algoritmo A\*. Ela inicializa a fila de prioridade com o nó inicial, e em cada iteração, remove o nó de menor custo da fila e expande os seus vizinhos, calculando os custos acumulados e heurísticas correspondentes. O processo continua até encontrar o nó de destino ou até que a fila de prioridade esteja vazia.

Resultados

Foram realizados três testes para buscar rotas da cidade de origem até Bucareste, usando o algoritmo A\* com a função heurística fornecida. Os resultados obtidos foram os seguintes:

Teste 1

Origem: Arad, Destino: Bucareste Caminho encontrado: Arad -> Sibiu -> Rimnicu Vilcea -> Pitesti -> Bucharest Custo acumulado: 418

Teste 2

Origem: Timisoara, Destino: Bucareste Caminho encontrado: Timisoara -> Arad -> Sibiu -> Rimnicu Vilcea -> Pitesti -> Bucharest Custo acumulado: 641

Teste 3

Origem: Craiova, Destino: Bucareste Caminho encontrado: Craiova -> Pitesti -> Bucharest Custo acumulado: 239

Discussão

O algoritmo A\* demonstrou ser eficiente na busca de rotas na Romênia, encontrando caminhos de menor custo com base na função heurística. A utilização da fila de prioridade permitiu a ordenação dos nós explorados de acordo com a função f(n), priorizando os nós com menor custo acumulado mais heurística estimada.

A função heurística fornecida foi utilizada para estimar a distância até o destino (Bucareste) a partir da cidade de origem. Essa função heurística considera a distância em linha reta entre cada cidade e Bucareste, conforme os valores fornecidos no grafo.

No entanto, é importante destacar que a função heurística utilizada nesse algoritmo é específica para o problema de busca de rotas até Bucareste. Ela não pode ser aplicada para buscar rotas até outras cidades de destino na Romênia. Caso seja necessário buscar rotas partindo de outras cidades, será necessário modificar a função heurística de acordo com o objetivo desejado.

Conclusão

O algoritmo A\* com fila de prioridade mostrou-se eficiente na busca de rotas na Romênia. Utilizando uma função heurística apropriada, foi capaz de encontrar caminhos de menor custo, levando em consideração tanto o custo acumulado como uma estimativa do custo restante até o destino.

É importante ressaltar que o sucesso do algoritmo depende da escolha adequada da função heurística, pois ela influencia diretamente no desempenho e qualidade dos resultados obtidos. Portanto, é recomendado estudar e ajustar a função heurística de acordo com o problema específico em questão.

Além disso, o algoritmo pode ser facilmente adaptado para buscar rotas partindo de outras cidades, desde que a função heurística correspondente seja ajustada para o novo objetivo.

Em resumo, o algoritmo A\* com fila de prioridade é uma poderosa ferramenta para busca de rotas, oferecendo uma combinação de eficiência e qualidade na solução encontrada.