

Relatório: Planejamento Inteligente de Drones para Monitoramento de Alagamentos

Gabriel Pietro Leone
Nº USP: 13874729
SCC0530 - Inteligência Artificial

May 30, 2025

Contents

1	Introdução	2
2	Descrição do Problema	2
3	Estratégia Adotada	2
4	Implementação do Algoritmo A*	2
4.1	Representação dos Dados	2
4.2	Estrutura do Nó	3
4.3	Função de Custo	3
4.4	Geração de Sucessores	3
4.5	Limite de Iterações	3
5	Resultados	3
5.1	Rota Planejada	3
5.2	Estatísticas da Missão	4
6	Justificativa das Escolhas	4
6.1	Algoritmo A*	4
6.2	Representação dos Dados	4
6.3	Visualização	4
7	Conclusão	4
8	Referências	4

1 Introdução

Este relatório apresenta uma solução para o planejamento inteligente de rotas de um drone de monitoramento que parte de uma capital brasileira e visita o maior número possível de cidades afetadas por alagamentos, respeitando uma autonomia de 750 km por trecho. O drone deve pousar em capitais para reabastecimento quando necessário, operando a uma velocidade constante de 100 km/h, sem considerar efeitos meteorológicos. O objetivo é maximizar a cobertura de cidades afetadas e minimizar as paradas para reabastecimento, utilizando o algoritmo A^* como base para a solução.

2 Descrição do Problema

O desafio consiste em planejar uma rota eficiente para um drone com as seguintes restrições e objetivos:

- **Ponto de Partida:** Uma capital brasileira (escolhida como Brasília neste caso).
- **Cidades-Alvo:** Cidades com registros de alagamentos, identificadas por coordenadas geográficas.
- **Autonomia:** Máximo de 750 km por trecho antes de reabastecer em uma capital.
- **Otimização:** Maximizar o número de cidades visitadas e minimizar o número de reabastecimentos.

A solução deve considerar a logística de reabastecimento e a cobertura geográfica, garantindo eficiência na missão.

3 Estratégia Adotada

A estratégia escolhida utiliza o algoritmo A^* , um método de busca informada que combina busca em largura com uma heurística para encontrar a rota ótima. O A^* foi adaptado para priorizar a maximização do número de cidades visitadas, seguida pela minimização do número de reabastecimentos e da distância total percorrida. Esta abordagem é ideal para problemas de planejamento de rotas com múltiplos objetivos e restrições.

4 Implementação do Algoritmo A^*

4.1 Representação dos Dados

Os dados foram representados como:

- **Cidades com Alagamentos:** Coordenadas geográficas (latitude e longitude) extraídas de um arquivo Excel.
- **Capitais:** Coordenadas geográficas de capitais brasileiras obtidas de um arquivo CSV.
- **Distâncias:** Calculadas com a fórmula de Haversine, que determina a distância em linha reta entre dois pontos na superfície terrestre.

4.2 Estrutura do Nó

Cada nó no A^* contém:

- **Posição Atual:** Nome e coordenadas da cidade ou capital.
- **Cidades Visitadas:** Conjunto de coordenadas das cidades já visitadas.
- **Custo Acumulado (g_score):** Tupla com (número negativo de cidades visitadas, número de reabastecimentos, distância total).
- **Caminho Percorrido:** Lista de passos realizados.
- **Distância no Segmento Atual:** Distância desde o último reabastecimento.
- **Número de Reabastecimentos:** Contagem de paradas para recarga.

4.3 Função de Custo

A função de custo (g_score) é uma tupla priorizando:

1. Maximizar cidades visitadas (valor negativo para minimização).
2. Minimizar reabastecimentos.
3. Minimizar distância total.

Essa hierarquia garante que a solução otimize os objetivos principais do problema.

4.4 Geração de Sucessores

Para cada nó, os sucessores são:

1. **Visitar Cidade Não Visitada:** Se a distância até a cidade estiver dentro da autonomia restante.
2. **Reabastecer em Capital:** Se a capital for alcançável e houver progresso na missão (evitando loops desnecessários).

4.5 Limite de Iterações

Um limite de 30.000 iterações foi estabelecido para evitar execução infinita, garantindo a praticidade da solução.

5 Resultados

5.1 Rota Planejada

A rota foi visualizada em um mapa interativo gerado com a biblioteca Folium, incluindo:

- Marcadores para a capital de partida, cidades visitadas e capitais de reabastecimento.
- Linhas de rota diferenciadas por tipo (monitoramento e reabastecimento).
- Informações detalhadas sobre cada ponto e segmento.

O mapa foi salvo como `rota_drone.html` para análise.

5.2 Estatísticas da Missão

Um relatório detalhado foi gerado em formato de texto, contendo:

- **Número de Cidades Visitadas:** Total de cidades monitoradas.
- **Número de Reabastecimentos:** Quantidade de paradas.
- **Distância Total:** Soma dos trechos percorridos.
- **Tempo Estimado:** Calculado com base na velocidade de 100 km/h.
- **Eficiência:** Médias como cidades por tanque e consumo por cidade.

O relatório foi salvo com um nome dinâmico, como `relatorio_rota_drone_[data].txt`.

6 Justificativa das Escolhas

6.1 Algoritmo A*

O A* foi selecionado por sua eficiência em encontrar soluções ótimas em problemas de busca com custos, especialmente com uma função de custo adaptada aos objetivos do problema.

6.2 Representação dos Dados

A fórmula de Haversine foi escolhida por sua precisão em calcular distâncias geográficas, essencial para respeitar a autonomia do drone.

6.3 Visualização

A biblioteca Folium foi utilizada para criar mapas interativos, facilitando a análise visual da rota e a comunicação dos resultados.

7 Conclusão

A solução desenvolvida utiliza o algoritmo A* para planejar eficientemente a rota de um drone de monitoramento, atendendo às restrições de autonomia e otimizando a cobertura de cidades afetadas por alagamentos. Os entregáveis, incluindo o código, mapa interativo e relatório estatístico, demonstram uma abordagem robusta e prática para o problema proposto.

8 Referências

- Documentação da biblioteca Folium: <https://python-visualization.github.io/folium/>
- Documentação da biblioteca Haversine: <https://github.com/mapado/haversine>

- Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). "A Formal Basis for the Heuristic Determination of Minimum Cost Paths".