

**UNIDADE CURRICULAR:** Introdução à Inteligência Artificial

**CÓDIGO:** 21071

**DOCENTE:** José Coelho

**A preencher pelo estudante**

**NOME:** Vítor Manuel Metrogos Frango

**N.º DE ESTUDANTE:** 1802925

**CURSO:** Licenciatura em Engenharia Informática

**DATA DE ENTREGA:** 20 maio 2024

**TRABALHO / RESOLUÇÃO:**

Este relatório apresenta a implementação e os resultados de um algoritmo de otimização aplicado a um problema em que o objetivo é minimizar o custo médio de deslocação das famílias para a estação mais próxima, ou seja, minimizar o número de estações (A) e o custo médio de deslocação (B), respeitando a fórmula de custo total definida no enunciado. O **estado inicial** são os territórios de NxM zonas (instâncias 1 a 20) com número de famílias em cada zona, havendo **ações possíveis** na escolha das posições para as estações em que o **modelo de transição** são todos os movimentos horizontais, verticais e diagonais para determinar a distância de cada zona até à estação mais próxima, respeitando os **custos** conforme as distâncias (também do enunciado). Os **sucessores** e o **espaço de estados** são todas a novas posições possíveis para as estações após cada ação e todas as combinações possíveis de posições para as estações. Por fim obtém se uma **solução** que será a implantação das estações conforme o cálculo do custo total tendo também em conta a **restrições** definidas para o problema

Para a resolução do problema optei pela escolha do algoritmo A\* melhorativo, no entanto este não seria necessariamente a única ou melhor escolha para todas as situações. A escolha foi adequada a diversos fatores entre eles a utilização preferencial sobre os lecionados na UC, mas outros me levaram à escolha do A\* melhorativo:

1. Problema de caminho: o problema de encontrar a melhor localização das estações pode ser modelado como um problema de caminho num grafo onde cada nó representa uma configuração de estações e as arestas representam a edição de uma nova estação
2. Heurística: o enunciado forneceu uma heurística clara para estimar o custo de uma solução o que é essencial para o bom desempenho de A\*
3. Otimização: A versão melhorativa do A\* procura iterativamente melhorar a solução o que pode set útil
4. Restrição do problema: A restrição do custo medio encaixa se bem na estratégia de busca do A\* melhorativo, que pode parar quando encontra uma solução que satisfaz essa restrição

Para a implementação em Python do algoritmo e de acordo com o solicitado, tive em linha de conta as seguintes configurações:

1. *custo\_dist*: um dicionário que define os custos de deslocamento unitários para diferentes distâncias
2. *max\_dist*: a distância máxima considerada para o cálculo do custo de deslocamento
3. *max\_time*: o tempo máximo de execução do algoritmo em msec (60000 ms = 1 minuto)
4. *max\_avaliacoes*: o número máximo de avaliações de soluções permitidas (100.000)
5. *peso\_estacao* e *peso\_custo*: os pesos utilizados na função heurística para equilibrar a importância do número de estações e do custo medo de deslocamento.

Resultados:



Screenshots com os outputs do código implementado em PyCharm 2024.1.1 (Professional Edition) Build #PY-241.15989.155, built on April 29, 2024 (há um delay de msec nos resultados em relação à tabela anterior em virtude os dados terem sido extraídos em momentos de execução diferentes)

|  |  |
| --- | --- |
| A screenshot of a computer  Description automatically generated |  |
|  |  |
|  | A screenshot of a computer  Description automatically generated |
|  |  |
|  |  |

Observações:

1. As instâncias com matrizes menores (IDs 1 a 10) conseguiram encontrar soluções dentro do tempo limite, enquanto as maiores (IDs 11 a 20) não conseguiram no tempo limite de 1 minuto.
2. O tempo de execução aumenta significativamente com o tamanho da matriz (território), evidenciado pelos tempos de execução das instâncias maiores que excedem 60seg
3. A maioria das soluções encontradas utilizou apenas uma solução (estação), exceto as instâncias maiores (IDs 7, 9 e 10) que utilizaram 2 ou 3 estações
4. O custo medio de deslocação (B) tende a aumentar com o tamanho e complexidade da matriz (território), indicando que a distancia media percorrida pelas famílias aumenta quando há mais zonas
5. O custo total (A + B) também aumenta com a complexidade do território refletindo a necessidade de mais estações ou maiores deslocamentos pata atender a todas as famílias

Conclusão:

O algoritmo A\* melhorativo implementado mostra se eficiente para territórios menores, encontrando para eles soluções de baixo custo e em tempo razoável, no entanto nos territórios maiores (matriz > 10) o tempo de execução e o número de avaliações limitaram a sua eficácia não conseguindo encontrar soluções validas dentro do limite de tempo de 1 minuto, isto sugere que melhorias ou otimizações adicionais podem ser necessárias para lidar com cenários mais complexos.

Portanto e pela análise dos resultados obtidos a execução do algoritmo A\* melhorativo revela tanto pontos fortes quanto limitações especialmente em relação à escalabilidade e eficiência em diferentes tamanhos e complexidades de territórios.

Algumas recomendações para melhorias futuras podem passar por utilização de heurísticas mais eficientes, implementação de paralelização, utilização de métodos de redução de espaço em busca (técnicas de podas para eliminar estados irrelevantes), utilização de combinação do algoritmo o A\* com outros algoritmos (hibridização)

Webgrafia:

* Russell, S. J., Norvig, P., & Davis, E. (2010). Artificial intelligence: A modern approach (3rd ed). Prentice Hall.
* Burke, E. K., & Kendall, G. (Eds.). (2014). Search Methodologies. Springer US. <https://doi.org/10.1007/978-1-4614-6940-7>
* <https://www.youtube.com/@wilsoncastello>
* <https://www.youtube.com/watch?v=-L-WgKMFuhE&t=1s>
* <https://www.youtube.com/watch?v=6TsL96NAZCo>
* <https://youtu.be/UwtjG1BUHJA?si=jK3j0oEGQ61ni4Dm>
* Co-pilot GitHub
* https://chat.openai.com