

”

E-fólio B | Folha de resolução para E-fólio



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 as 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 as **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 ser 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:

- a) *custo_dist*: um dicionário que define os custos de deslocamento unitários para diferentes distâncias
- b) *max_dist*: a distância máxima considerada para o cálculo do custo de deslocamento
- c) *max_time*: o tempo máximo de execução do algoritmo em msec (60000 ms = 1 minuto)
- d) *max_avaliacoes*: o número máximo de avaliações de soluções permitidas (100.000)
- e) *peso_estacao* e *peso_custo*: os pesos utilizados na função heurística para equilibrar a importância do número de estações e do custo medio de deslocamento.

Resultados:

Instancias	ID1	ID2	ID3	ID4	ID5	ID6	ID7	ID8	ID9	ID10
Avaliações	26	26	50	50	82	82	122	122	170	170
Gerações	25	25	49	49	81	81	121	121	169	169
Custos	1079	1078	1104	1121	1237	1233	2228	1269	3264	3186
Tempo (msec)	34	34	409	412	2825	2795	13347	13573	49834	49475
Melhor resultado	1079	1078	1104	1121	1237	1233	2228	1269	3264	3186

Instancias	ID11	ID12	ID13	ID14	ID15	ID16	ID17	ID18	ID19	ID20
Avaliações	Sem Soluções dentro do limite de 1 minuto									
Gerações										
Custos										
Tempo (msec)										
Melhor resultado										

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)

```
"/Users/vitorfrango/Library/CloudStorage/OneDrive
-----
Instancia ID 1:
0 7 0 0 4
0 0 0 4 0
1 0 0# 0 0
4 4 1 0 0
6 0 3 4 4
-----
Avaliações: 26
Gerações: 25
Custo: 1079
Tempo: 33 msec
-----
Número de estações (A): 1
Custo médio de deslocação (B): 0.786
-----
Instancia ID 2:
4 0 0 10 1
1 0 0 0 0
0 0 1# 6 3
0 4 0 0 2
8 0 6 3 0
-----
Avaliações: 26
Gerações: 25
Custo: 1078
Tempo: 32 msec
-----
Número de estações (A): 1
Custo médio de deslocação (B): 0.776
-----
```

```
-----
Instancia ID 3:
0 8 0 4 5 10 0
0 4 0 7 0 4 0
0 2 4 2# 0 0 2
0 7 0 1 2 0 0
2 4 0 0 3 0 2
0 4 0 0 3 0 0
2 0 0 0 0 0 0
-----
Avaliações: 50
Gerações: 49
Custo: 1104
Tempo: 406 msec
-----
Número de estações (A): 1
Custo médio de deslocação (B): 1.037
-----
Instancia ID 4:
0 0 1 0 7 0 1
0 1 4 0 0 0 4
0 0 0 0 2 0 0
3 1 0 8 5# 7 7
0 4 0 3 0 0 0
0 0 0 3 2 4 2
0 8 3 6 3 0 0
-----
Avaliações: 50
Gerações: 49
Custo: 1121
Tempo: 423 msec
-----
Número de estações (A): 1
Custo médio de deslocação (B): 1.213
-----
```

```
-----  
Instancia ID 5:  
6 7 2 0 0 0 0 0 0  
3 3 6 0 8 4 3 1 0  
0 0 8 0 0 0 2 4 0  
0 0 0 1 0 3 2 0 0  
0 0 0 7# 4 0 1 0 0  
12 8 0 5 4 1 4 3 4  
8 0 1 2 4 3 3 0 0  
1 1 0 0 0 0 5 0 0  
4 0 0 0 4 6 0 13 2  
-----
```

```
Avaliações: 82  
Gerações: 81  
Custo: 1237  
Tempo: 2949 msec  
-----
```

```
Número de estações (A): 1  
Custo médio de deslocação (B): 2.370  
-----
```

```
-----  
Instancia ID 6:  
0 0 0 0 0 0 0 0 0  
4 0 8 4 0 0 0 0 0  
0 0 0 0 0 0 0 0 0  
0 0 0 0 3 0 0 1 0  
0 3 0 0 0 0 0 0 0  
0 0 0 1 1# 0 0 3 0  
0 0 2 4 0 0 0 1 0  
0 2 0 0 8 0 4 3 10  
0 0 3 0 0 4 0 0 0  
-----
```

```
Avaliações: 82  
Gerações: 81  
Custo: 1233  
Tempo: 2779 msec  
-----
```

```
Número de estações (A): 1  
Custo médio de deslocação (B): 2.333  
-----
```

```
-----  
Instancia ID 7:  
0 0 0 0 0 3 0 0 0 0 0  
0 0 11 2 0 0 9 3 0 0 3  
0 0 0 3 1 0 2 0 0 0 0  
4 1 2 3# 0 4 0 0 4 0 0  
5 0 0 0 4 0 1 0 4 3 0  
0 0 0 7 4 0 1 0 0 7 0  
0 8 0 0 0 0# 3 0 1 0 3  
0 3 0 0 5 2 3 0 0 0 2  
0 0 0 3 1 0 2 8 0 0 0  
0 3 4 0 7 0 0 7 0 0 0  
4 2 0 4 0 3 0 0 5 7 0  
-----
```

```
Avaliações: 122  
Gerações: 121  
Custo: 2228  
Tempo: 13645 msec  
-----
```

```
Número de estações (A): 2  
Custo médio de deslocação (B): 2.276  
-----
```

```
-----  
Instancia ID 8:  
1 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 0  
0 0 10 10 0 0 0 4 5 0 0  
0 4 1 0 8 0 0 0 0 0 5  
8 0 0 0 0 0 6 0 0 0 0  
0 0 0 0 13# 0 0 0 2 0 3  
0 0 0 0 4 0 0 0 0 1 0  
0 0 0 0 0 0 0 0 0 0 0  
0 0 4 0 0 0 0 3 0 0 0  
4 1 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 0  
-----
```

```
Avaliações: 122  
Gerações: 121  
Custo: 1269  
Tempo: 13853 msec  
-----
```

```
Número de estações (A): 1  
Custo médio de deslocação (B): 2.691  
-----
```

```

-----
Instancia ID 9:
2 4 0 0 6 7 3 4 0 0 3 0 1
0 0 2 0 3 0 0 6 0 0 8 11 3
0 3 0 8 0 0 2 0 0 0 0 0 4
2 0 0 0 0# 0 0 0 0 3 2 0 0
0 6 0 8 0 3 0 0 0 0 0 0 1
0 3 0 2 0 0 9 0 0 0 0 5 6
1 9 4 0 0 2 4 0 0 0 3 2 0
2 3 0 4 0 0 0 6 2 0 1 0 3
0 0 0 0 0 6 0 0 0# 2 2 0 8
7 2 4 2 0 0 6 4 1 0 0 0 7
0 0 0 11# 0 0 0 0 3 4 0 9 0
0 0 0 0 1 4 3 4 0 0 0 3 11
0 0 4 7 7 0 0 2 0 2 5 0 1
-----

Avaliações: 170
Gerações: 169
Custo: 3264
Tempo: 51075 msec
-----

Número de estações (A): 3
Custo médio de deslocação (B): 2.640
-----

```

```

-----
Instancia ID 10:
0 0 1 4 0 0 9 0 0 0 12 0 1
0 0 0 0 0 0 0 0 0 0 1 0 0 0
1 0 0 0 0 0 2 0 0 2 0 0 0
0 0 0 0 0 9 4 0 0 0 6 0 0
0 6 9 0 0 0 0# 0 0 0 0 0 0
0 0 0 0 0 0 0 1 6 10 0 1 4
0 3 0# 0 0 1 0 0 0 0 0 2 0
0 0 0 1 3 0 0 0 0 9 0 0 0
9 0 0 3 3 0 0 0 0 3# 4 0 0
0 1 4 0 0 0 0 0 0 5 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0
2 0 0 0 0 3 3 0 0 0 0 0 10
0 0 0 0 0 0 0 0 0 4 0 0 0
-----

Avaliações: 170
Gerações: 169
Custo: 3186
Tempo: 52581 msec
-----

Número de estações (A): 3
Custo médio de deslocação (B): 1.859
-----

```

```

-----
Instancia ID 11: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 119
Gerações: 119
Tempo de execução: 60477 msec
-----

Instancia ID 12: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 121
Gerações: 121
Tempo de execução: 60154 msec
-----

Instancia ID 13: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 97
Gerações: 97
Tempo de execução: 60508 msec
-----

Instancia ID 14: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 94
Gerações: 94
Tempo de execução: 61089 msec
-----

Instancia ID 15: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 80
Gerações: 80
Tempo de execução: 60556 msec
-----

```

```

-----
Instancia ID 16: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 82
Gerações: 82
Tempo de execução: 61225 msec
-----

Instancia ID 17: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 67
Gerações: 67
Tempo de execução: 60485 msec
-----

Instancia ID 18: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 69
Gerações: 69
Tempo de execução: 60678 msec
-----

Instancia ID 19: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 59
Gerações: 59
Tempo de execução: 61325 msec
-----

Instancia ID 20: Nenhuma solução válida encontrada em 1 minuto
Avaliações: 60
Gerações: 60
Tempo de execução: 60978 msec
-----

```

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 para 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>