

Investigação Operacional - Trabalho Prático 1

Carlos Machado a97114, Gustavo Pereira a96867, Vasco Oliveira a96361,
Cláudio Bessa a97063, Tiago Oliveira a97254

Universidade do Minho

1 Formulação do Problema

O problema em causa surge quando um drone precisa de inspecionar linhas de transporte de energia eléctrica em alta tensão para verificar se há vegetação a interferir com as linhas. Este pode percorrer as arestas (linhas de alta tensão) em qualquer sentido, tendo também possibilidade de viajar pelo ar, *i.e.*, sem seguir as linhas de alta tensão.

O objetivo principal deste desafio é fazer com que o drone percorra a menor distância possível, partindo do vértice I e acabando no vértice F, sendo obrigatório percorrer todas as linhas pelo menos uma vez, quer as que estão representadas originalmente no mapa, quer as linhas adicionaremos de modo a formar um caminho euleriano.

O maior número de estudante do grupo é 97254, o que significa que teremos de retirar as arestas C e E do mapa inicial, resultando no mapa que se observa na Figura 1.

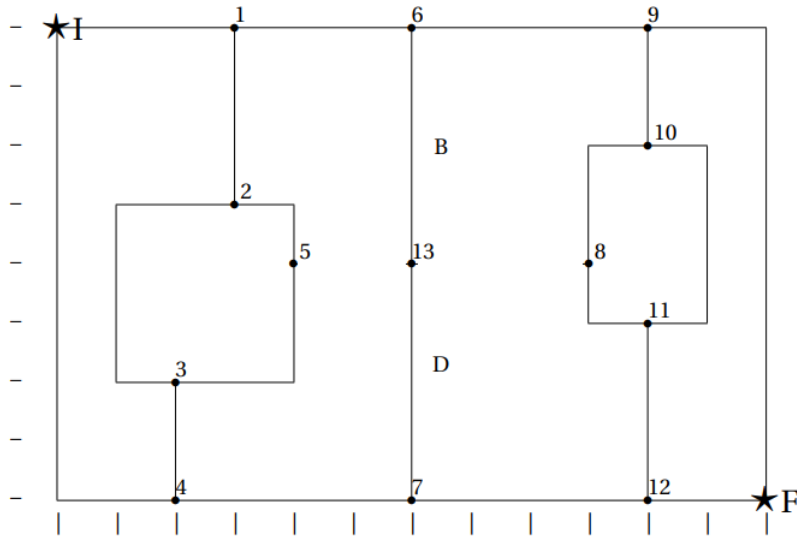


Figura 1. Mapa de linhas de alta tensão, após remoção das arestas C e E.

2 Modelo

Uma vez que pretendemos percorrer todas as arestas do percurso de forma eficiente, podemos observar este problema como um caminho euleriano, onde pretendemos encontrar um percurso para um caminho de forma a passar em todas as arestas uma só vez. Este caminho apenas existe quando os vértices interiores ao mapa têm grau de entrada/saída par, e os vértices que representam o início (I) e o fim (F) do percurso grau ímpar.

Na Figura 2, encontram-se assinalados os vértices onde é necessário aumentar o grau, *i.e.*, criar ou duplicar novas adjacências, para tornar o percurso um caminho euleriano. Desta forma, obtemos um problema de programação linear onde procuramos adicionar arestas tentando otimizar o itinerário de modo a minimizar a distância percorrida pelo drone.

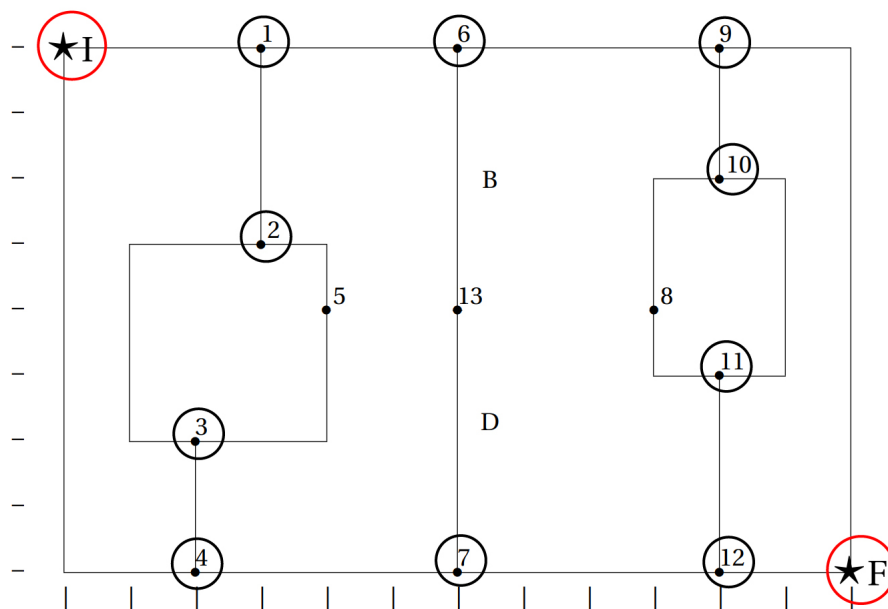


Figura 2. Vértices com grau insuficiente.

2.1 Variáveis de decisão

x_{ij} : existe ou não uma aresta a unir os vértices $i, j \in \{I, 1, 2, 3, 4, 6, 7, 9, 10, 11, 12, F\}$, onde $x_{ij} \in \{0, 1\}$.

```

53 bin xI1,xI2,xI3,xI4,xI6,xI7,xI9,xI10,xI11,xI12,xIF,
54 x12,x13,x14,x16,x17,x19,x110,x111,x112,x1F,
55 x23,x24,x26,x27,x29,x210,x211,x212,x1F,
56 x34,x36,x37,x39,x310,x311,x312,x3F,
57 x46,x47,x49,x410,x411,x412,x4F,
58 x67,x69,x610,x611,x612,x6F,
59 x79,x710,x711,x712,x7F,
60 x910,x911,x912,x9F,
61 x1011,x1012,x10F,
62 x1112,x11F,
63 x12F;

```

Figura 3. Variáveis de decisão

2.2 Parâmetros

d_{ij} : comprimento da aresta que une os vértices i e j . Os comprimentos de todas as arestas podem ser calculados a partir da seguinte tabela:

		x	0	3	3	2	2	4	6	6	9	10	10	10	10	6	12
		y	8	8	5	2	0	4	8	0	4	8	6	3	0	4	0
x	y		1	2	3	4	5	6	7	8	9	10	11	12	13	F	
0	8	1	0,00	3,00	4,24	6,32	8,25	5,66	6,00	10,00	9,85	10,00	10,20	11,18	12,81	7,21	14,42
3	8	1		0,00	3,00	6,08	8,06	4,12	3,00	8,54	7,21	7,00	7,28	8,60	10,63	5,00	12,04
3	5	2			0,00	3,16	5,10	1,41	4,24	5,83	6,08	7,62	7,07	7,28	8,60	3,16	10,30
2	2	3				0,00	2,00	2,83	7,21	4,47	7,28	10,00	8,94	8,06	8,25	4,47	10,20
2	0	4					0,00	4,47	8,94	4,00	8,06	11,31	10,00	8,54	8,00	5,66	10,00
4	4	5						0,00	4,47	4,47	5,00	7,21	6,32	6,08	7,21	2,00	8,94
6	8	6							0,00	8,00	5,00	4,00	4,47	6,40	8,94	4,00	10,00
6	0	7								0,00	5,00	8,94	7,21	5,00	4,00	4,00	6,00
9	4	8									0,00	4,12	2,24	1,41	4,12	3,00	5,00
10	8	9										0,00	2,00	5,00	8,00	5,66	8,25
10	6	10											0,00	3,00	6,00	4,47	6,32
10	3	11												0,00	3,00	4,12	3,61
10	0	12													0,00	5,66	2,00
6	4	13														0,00	7,21
12	0	F															0,00

Figura 4. Distancias Euclidianas fornecidas

2.3 Função objetivo

Minimização da distância percorrida pelo *drone*.

$$\min : \sum_{i,j} \in V \setminus \{5, 8, 13\}, d_{ij} \times x_{ij}, i < j$$

De se lembrar que não são incluídas as arestas que incluem os vértices 5, 8 e 13 na função objetivo, uma vez que estas não são contempladas no nosso mapa, pois se adicionássemos arestas a estes vértices eles paravam de ter grau par, acabando por não ter um caminho euleriano como resultado.

```

1 /* Objective function */
2 min: 3.00 xI1 + 4.24 xI2 + 6.32 xI3 + 8.25 xI4 + 6.00 xI6 + 10.00 xI7 + 10.00 xI9 + 10.20 xI10 + 11.18 xI11 + 12.81 xI12 + 14.42 xIF +
3 3.00 xI2 + 6.08 xI3 + 8.06 xI4 + 3.00 xI6 + 8.54 xI7 + 7.00 xI9 + 7.28 xI10 + 8.60 xI11 + 10.63 xI12 + 12.04 xIF +
4 3.16 x23 + 5.10 x24 + 4.24 x26 + 5.83 x27 + 7.62 x29 + 7.07 x210 + 7.28 x211 + 8.60 x212 + 10.30 x2F +
5 2.00 x34 + 7.21 x36 + 4.47 x37 + 10.00 x39 + 8.94 x310 + 8.06 x311 + 8.25 x312 + 10.20 x3F +
6 8.94 x46 + 4.00 x47 + 11.31 x49 + 10.00 x410 + 8.54 x411 + 8.00 x412 + 10.00 x4F +
7 8.00 x67 + 4.00 x69 + 4.47 x610 + 6.40 x611 + 8.94 x612 + 10.00 x6F +
8 8.94 x79 + 7.21 x710 + 5.00 x711 + 4.00 x712 + 6.00 x7F +
9 2.00 x910 + 5.00 x911 + 8.00 x912 + 8.25 x9F +
10 3.00 x1011 + 6.00 x1012 + 6.32 x10F +
11 3.00 x1112 + 3.61 x11F +
12 2.00 x12F;

```

Figura 5. Função objetivo

2.4 Restrições

Cada vértice apenas deve fazer parte de uma só aresta.

$\forall i \in V \setminus \{5, 8, 13\} \wedge j \in V \setminus \{5, 8, 13\}, \sum x_{ij} = 1, i < j$

```

15 /* Variable bounds */
16
17 RI: xI1 + xI2 + xI3 + xI4 + xI6 + xI7 + xI9 + xI10 + xI11 + xI12 + xIF = 1;
18 R1: xI1 + xI2 + xI3 + xI4 + xI6 + xI7 + xI9 + xI10 + xI11 + xI12 + xIF = 1;
19 R2: xI2 + xI2 + x23 + x24 + x26 + x27 + x29 + x210 + x211 + x212 + x2F = 1;
20 R3: xI3 + xI3 + x23 + x34 + x36 + x37 + x39 + x310 + x311 + x312 + x3F = 1;
21 R4: xI4 + xI4 + x24 + x34 + x46 + x47 + x49 + x410 + x411 + x412 + x4F = 1;
22 R6: xI6 + xI6 + x26 + x36 + x46 + x67 + x69 + x610 + x611 + x612 + x6F = 1;
23 R7: xI7 + xI7 + x27 + x37 + x47 + x67 + x79 + x710 + x711 + x712 + x7F = 1;
24 R9: xI9 + xI9 + x29 + x39 + x49 + x69 + x79 + x910 + x911 + x912 + x9F = 1;
25 R10: xI10 + xI10 + x210 + x310 + x410 + x610 + x710 + x910 + x1011 + x1012 + x10F = 1;
26 R11: xI11 + xI11 + x211 + x311 + x411 + x611 + x711 + x911 + x1011 + x1112 + x11F = 1;
27 R12: xI12 + xI12 + x212 + x312 + x412 + x612 + x712 + x912 + x1012 + x1112 + x12F = 1;
28 RF: xIF + x1F + x2F + x3F + x4F + x6F + x7F + x9F + x10F + x11F + x12F = 1;

```

Figura 6. Restrições

3 Solução ótima

Introduzimos os modelo criado no LPSolve e descobrimos as arestas a adicionar no percurso: $I - 1; 2 - 6; 3 - 4; 7 - 11; 9 - 10; 12 - F$. O comprimento total das arestas seleccionadas é de $18.24(u.c)$ sendo o comprimento total de $86.79(u.c)$.

Variables	MILP ...	re... ▼
	18.24	18.24
x11	1	1
x26	1	1
x34	1	1
x711	1	1
x910	1	1
x12F	1	1
x12	0	0

Figura 7. Output retornado pelo LPSolve

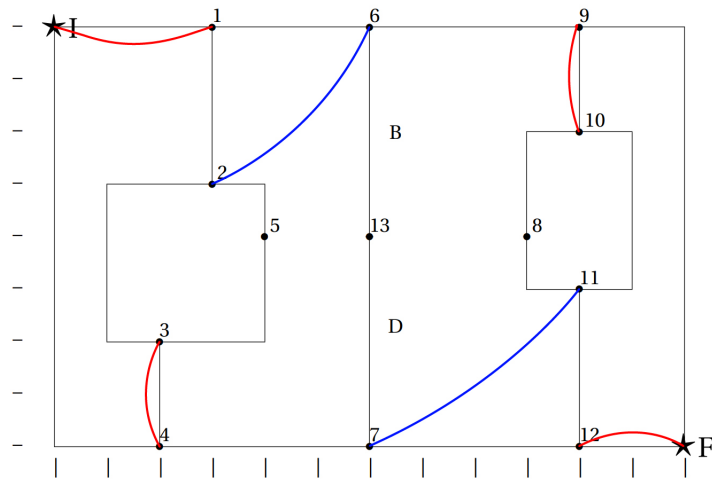


Figura 8. Arestas contempladas como solução ótima

Exemplo de percurso:
 $I \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow I \rightarrow 1 \rightarrow 6 \rightarrow 2 \rightarrow 5 \rightarrow 3 \rightarrow 4 \rightarrow 7 \rightarrow 13 \rightarrow 6 \rightarrow 9 \rightarrow$
 $10 \rightarrow 8 \rightarrow 11 \rightarrow 7 \rightarrow 12 \rightarrow F \rightarrow 12 \rightarrow 11 \rightarrow 10 \rightarrow 9 \rightarrow F$

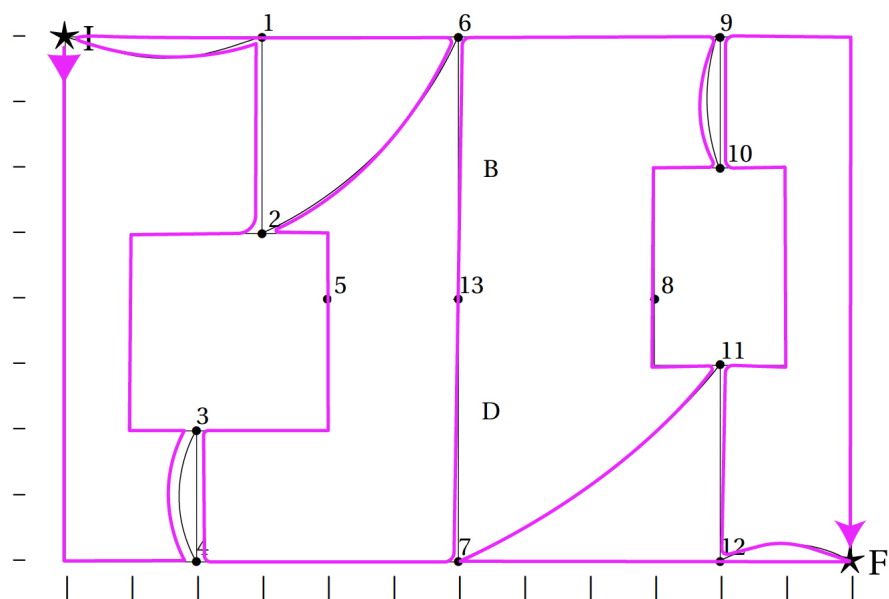


Figura 9. Exemplo de um possível percurso.

4 Validação do modelo

Mesmo confiando no método utilizado para obter a solução, é necessária uma validação do modelo utilizado. Para isso, recorreremos novamente ao LPSolve.

Vimos como um bom método de validação ir removendo, ao nosso modelo, as arestas pertencentes à solução ótima obtida anteriormente uma a uma, obrigando, desta forma, a gerar um novo percurso. Observando o valor dos outputs destas novas soluções, verifica-se que nenhuma delas apresenta um trajeto com comprimento menor do que o obtido no modelo correspondente à solução ótima.

Removendo a aresta *I-1*:

Arestas da solução: $I - 2; 1 - 6; 3 - 4; 7 - 11; 9 - 10; 12 - F$

Comprimento total das arestas selecionadas: 18.24 (*u.c.*)

Removendo a aresta *2-6*:

Arestas da solução: $I - 2; 1 - 6; 3 - 4; 7 - 11; 9 - 10; 12 - F$

Comprimento total das arestas selecionadas: 18.24(*u.c.*)

Removendo a aresta *3-4*:

Arestas da solução: $I - 1; 2 - 3; 4 - 7; 6 - 9; 10 - 11; 12 - F$

Comprimento total das arestas selecionadas: 19.16(*u.c.*)

Removendo a aresta *7-11*:

Arestas da solução: $I - 1; 2 - 6; 3 - 4; 9 - 10; 7 - 12; 11 - F$

Comprimento total das arestas selecionadas: 18.85(*u.c.*)

Removendo a aresta *9-10*:

Arestas da solução: $I - 1; 2 - 3; 4 - 7; 6 - 9; 10 - 11; 12 - F$

Comprimento total das arestas selecionadas: 19.16(*u.c.*)

Removendo a aresta *12-F*:

Arestas da solução: $I - 1; 2 - 6; 3 - 4; 7 - 12; 9 - 10; 11 - F$

Comprimento total das arestas selecionadas: 18.85(*u.c.*)