Universidade do Minho

Departamento de Informática

MDIO - TP2

Grupo Nº 21

Gonçalo Almeida (A84610)

Emanuel Rodrigues (A84776)

Lázaro Pinheiro (A86788)

Luís Ferreira (A76936)

29 de Novembro de 2019

Conteúdo

Capítulo 1

Introdução

O presente trabalho prático desenvolve-se no âmbito da Unidade Curricular Modelos Determinísticos de Investigação Operacional, lecionada no 1º semestre do 3º ano do curso de Engenharia Informática. O objetivo dos trabalhos práticos é desenvolver a capacidade de analisar sistemas complexos, de criar modelos para os descrever, de obter soluções para esses modelos utilizando programas computacionais adequados, de validar os modelos obtidos, de interpretar as soluções obtidas, e de elaborar recomendações para o sistema em análise.

No decorrer deste trabalho, procurou-se formular um modelo de transportes com vista a obter uma solução ótima.

Capítulo 2

Problema

O propósito deste trabalho prático é a determinação do circuito, ou conjunto de circuitos, em que todos os arcos de um grafo são percorridos, pelo menos, uma vez, minimizando a distância total percorrida. Note-se que pode ser necessário atravessar o mesmo arco mais de uma vez. Cada arco (rua) representa a ligação entre dois nodos, de um grafo, ou seja, o mapa de uma zona de uma cidade.

2.1 Parte 1

Para melhor compreensão do modelo decidiu-se associar a cada nodo um número, de modo a facilitar a associação de cada variável ao arco correspondente.

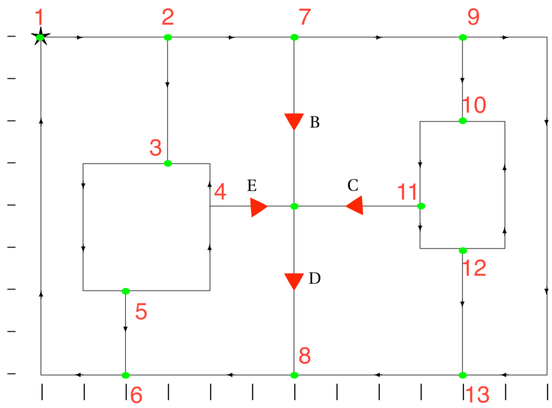


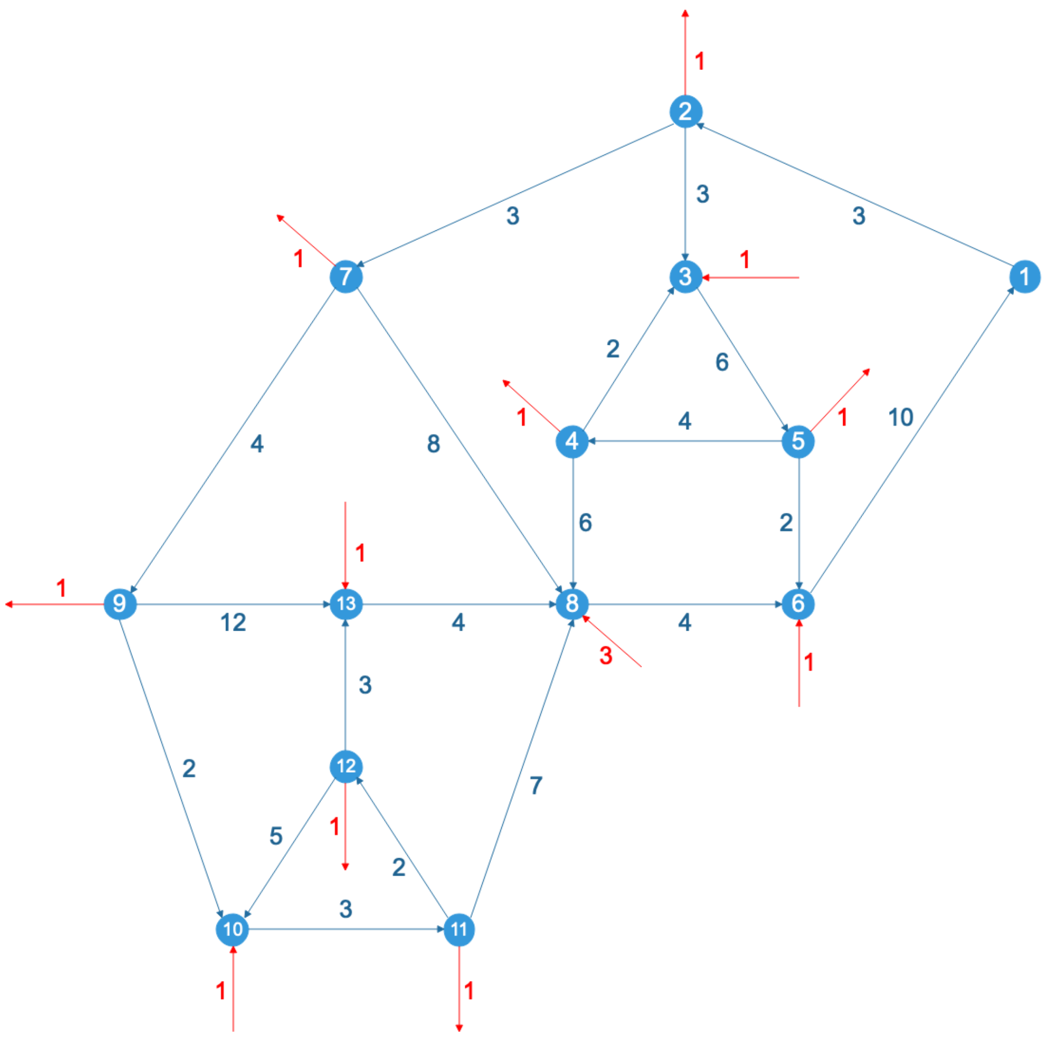
Figura 2.1: Nodos

Para a realização deste trabalho prático tomou-se como ponto de partida o modelo desenvolvido no primeiro trabalho prático, que consiste num modelo que minimiza a distância percorrida na recolha do lixo, passando por todas as ruas, com um único sentido (Problema do carteiro chinês). Este apresentava dois grupos distintos de restrições, que pretendiam assegurar que:

- o fluxo que entra num vértice do grafo deve ser igual ao que sai, para o percurso ser fechado.
- o fluxo em qualquer arco deve ser, pelo menos, uma unidade, para visitar todos os arcos.

O grupo das segundas restrições do modelo concebido no primeiro trabalho prático é limitado inferiormente. Como o software apenas aceita arcos com limites superiores procedeu-se à mudança de variável $y_{ij} = x_{ij} - l_{ij}$, $\forall(i,j) \in A$, em que l_{ij} é o limite inferior de fluxo no arco (i,j).

2.1.1 Grafo



2.1.2 Variáveis de decisão

Cada variável de decisão representa o fluxo, entre dois nodos, do grafo.

$y_{ij} \in \mathbb{N}$: fluxo no arco com origem no nodo $i \in \mathbb{N}$ e destino no nodo $j \in \mathbb{N}$.

2.1.3 Dados

Seja c_{ij} o custo unitário de transporte no arco orientado (i,j) que, neste caso, é a distância associada ao arco (i,j). Então temos os seguintes custos:

$c_{12} = 3$	$c_{48} = 6$	$c_{79} = 4$	$c_{118} = 7$
$c_{23} = 3$	$c_{54} = 4$	$c_{86} = 4$	$c_{1112} = 2$
$c_{27} = 3$	$c_{56} = 2$	$c_{910} = 2$	$c_{1210} = 5$
$c_{35} = 6$	$c_{61} = 10$	$c_{913} = 12$	$c_{1213} = 3$
$c_{43} = 2$	$c_{78} = 8$	$c_{1011} = 3$	$c_{138} = 4$

Seja b_j a oferta (valor positivo) ou procura (valor negativo) no vértice j. Então temos os seguintes valores:

$$
\begin{array}{lllll}
b_1 = 0 & b_4 = -1 & b_7 = -1 & b_{10} = 1 & b_{13} = 1 \\
b_2 = -1 & b_5 = -1 & b_8 = 3 & b_{11} = -1 & \\
b_3 = 1 & b_6 = 1 & b_9 = -1 & b_{12} = -1 &
\end{array}
$$

2.1.4 Restrições

Para não se tornar maçador, procurou-se generalizar a justificação das restrições apresentadas.

O seguinte grupo de restrições é designado por restrições de capacidade, que garantem que o fluxo no arco orientado (i,j) é igual ou superior a 0 e menor ou igual do que a capacidade do arco, neste caso virtualmente infinita (100) para todos os arcos.

$$
\begin{array}{lllll}
y_{12} \geq 0 & y_{43} \geq 0 & y_{61} \geq 0 & y_{910} \geq 0 & y_{1210} \geq 0 \\
y_{23} \geq 0 & y_{48} \geq 0 & y_{78} \geq 0 & y_{913} \geq 0 & y_{1213} \geq 0 \\
y_{27} \geq 0 & y_{54} \geq 0 & y_{79} \geq 0 & y_{1011} \geq 0 & y_{138} \geq 0 \\
y_{35} \geq 0 & y_{56} \geq 0 & y_{86} \geq 0 & y_{118} \geq 0 & y_{1112} \geq 0
\end{array}
$$

O seguinte grupo de restrições é designado por restrições de conservação de fluxo e garantem que o somatório de todos os fluxos no arco orientado (i,j) somado com o somatório de todos os fluxos no arco orientado (j,i) é igual à oferta ou procura no vértice j.

$$
\begin{array}{llll}
y_{12} + 1 = y_{61} + 1 & y_{35} + 1 = y_{54} + y_{56} + 2 & y_{86} + 1 = y_{48} + y_{78} + y_{118} + y_{138} + 4 & y_{12} + 1 = y_{23} + y_{27} + 2 \\
y_{54} + 1 = y_{43} + y_{48} + 2 & y_{1011} + 1 = y_{910} + y_{1210} + 2 & y_{138} + 1 = y_{913} + y_{1213} + 2 & y_{27} + 1 = y_{78} + y_{79} + 2 \\
y_{61} + 1 = y_{56} + y_{86} + 2 & y_{1011} + 1 = y_{118} + y_{1112} + 2 & y_{35} + 1 = y_{23} + y_{43} + 2 & y_{79} + 1 = y_{910} + y_{913} + 2 \\
y_{1112} + 1 = y_{1210} + y_{1213} + 2 & & &
\end{array}
$$

2.1.5 Função objetivo

$$
\begin{aligned}
min\ z = {} & 3y12 + 3y23 + 3y27 + 6y35 + 2y43 + 6y48 + 4y54 + 2y56 + 10y61 + 8y78 + 4y79 \\
& + 4y86 + 2y910 + 12y913 + 3y1011 + 7y118 + 2y1112 + 5y1210 + 3y1213 + 4y138 + 93
\end{aligned}
$$

O coeficiente de cada variável y_{ij} representa o comprimento do respetivo arco. Substituindo cada valor da solução ótima na função objetivo, obtém-se a distância associada ao menor percurso.

2.1.6 Solução ótima

Neste problema a solução ótima é obtida, quando todos os arcos de um conjunto de circuitos de um grafo são percorridos, pelo menos uma vez, minimizando o fluxo de cada arco orientado.

$$y^* = \begin{pmatrix} y12* \\ y23* \\ y27* \\ y35* \\ y43* \\ y48* \\ y54* \\ y56* \\ y61* \\ y78* \\ y79* \\ y86* \\ y910* \\ y913* \\ y1011* \\ y118* \\ y1112* \\ y1210* \\ y1213* \\ y138* \end{pmatrix} = \begin{pmatrix} 5 \\ 1 \\ 3 \\ 2 \\ 0 \\ 0 \\ 1 \\ 0 \\ 5 \\ 0 \\ 2 \\ 4 \\ 1 \\ 0 \\ 2 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

A partir resolução do problema de rede de transportes recorrendo ao software RELAX4, obteve-se a solução ótima de valor 131, ao qual se somou 93, valor correspondente a passar em cada aresta uma vez, obtendo-se a seguinte solução ótima.

$$Z(y^*) = 224;$$

Considera-se que o caminho pertence ao conjunto dos caminhos de reposicionamento, se existe um fluxo não negativo entre um vértice de excesso e um vértice de defeito.

2.1.7 Validação de resultados

De modo a validar o resultado obtido, substitui-se os valores da solução ótima nas restrições, como exemplificado abaixo, obtendo afirmações verdadeiras.

y12 >= 0 <=> 5 >= 0
y23 >= 0 <=> 1 >= 0
y27 >= 0 <=> 3 >= 0
y35 >= 0 <=> 2 >= 0
y43 >= 0 <=> 0 >= 0
y48 >= 0 <=> 0 >= 0
y54 >= 0 <=> 1 >= 0
y56 >= 0 <=> 0 >= 0
y61 >= 0 <=> 5 >= 0
y78 >= 0 <=> 0 >= 0
y79 >= 0 <=> 2 >= 0
y86 >= 0 <=> 4 >= 0
y910 >= 0 <=> 1 >= 0
y913 >= 0 <=> 0 >= 0
y1011 >= 0 <=> 2 >= 0
y118 >= 0 <=> 0 >= 0
y1112 >= 0 <=> 1 >= 0
y1210 >= 0 <=> 0 >= 0
y1213 >= 0 <=> 0 >= 0
y138 >= 0 <=> 1 >= 0

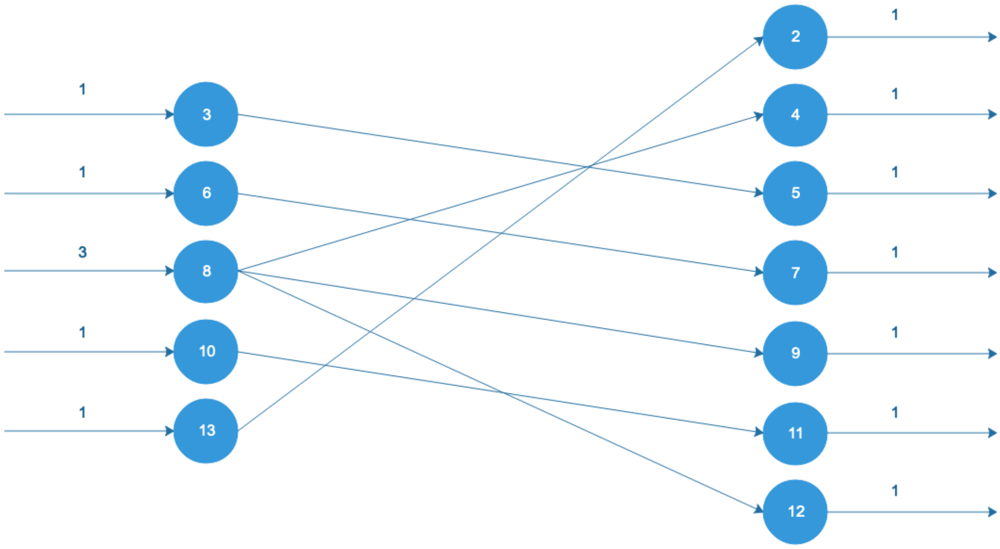
y12 + 1 = y61 + 1 <=> 5 + 1 = 5 + 1 <=> 6 = 6
y12 + 1 = y27 + y23 + 2 <=> 5 + 1 = 3 + 1 + 2 <=> 6 = 6
y27 + 1 = y78 + y79 + 2 <=> 3 + 1 = 0 + 2 + 2 <=> 4 = 4
y35 + 1 = y23 + y43 + 2 <=> 2 + 1 = 1 + 0 + 2 <=> 3 = 3
y35 + 1 = y56 + y54 + 2 <=> 2 + 1 = 0 + 1 + 2 <=> 3 = 3
y54 + 1 = y43 + y48 + 2 <=> 1 + 1 = 0 + 0 + 2 <=> 2 = 2
y61 + 1 = y86 + y56 + 2 <=> 5 + 1 = 4 + 0 + 2 <=> 6 = 6
y79 + 1 = y910 + y913 + 2 <=> 2 + 1 = 1 + 0 + 2 <=> 3 = 3
y86 + 1 = y78 + y138 + y48 + y118 + 4 <=> 4 + 1 = 0 + 1 + 0 + 0 + 4 <=> 5 = 5
y1011 + 1 = y910 + y1210 + 2 <=> 2 + 1 = 1 + 0 + 2 <=> 3 = 3
y1011 + 1 = y118 + y1112 + 2 <=> 2 + 1 = 0 + 1 + 2 <=> 3 = 3
y1112 + 1 = y1213 + y1210 + 2 <=> 1 + 1 = 0 + 0 + 2 <=> 2 = 2
y138 + 1 = y913 + y1213 + 2 <=> 1 + 1 = 0 + 0 + 2 <=> 2 = 2

2.2 Parte 2

Para além da formulação apresentada anteriormente, existe uma outra formulação para este problema que revela a estrutura da solução ótima. Consiste em considerar variáveis de decisão x_{ij} , cada uma correspondendo a um caminho entre um vértice de excesso i e um vértice de defeito j , como se explica de seguida.

Quando, num cruzamento, o número de ruas que entram é maior do que o número de ruas que saem (vértice de excesso), torna-se necessário repetir, pelo menos, uma das ruas de saída. O mesmo se passa quando o número de ruas que entram é menor do que o número de ruas que saem (vértice de defeito): torna-se necessário repetir, pelo menos, uma das ruas de entrada. Estes arcos repetidos devem fazer parte de caminhos adicionais que saem dos vértices de excesso e chegam aos vértices de defeito, dando origem a percursos válidos. O modelo de minimização consiste em determinar os caminhos adicionais a realizar de modo a minimizar a soma dos seus custos.

2.2.1 Grafo Bipartido



2.2.2 Problema

O problema é balanceado, visto que a soma das procuras e ofertas de todos os vértices é 0. Isto deve-se ao facto de não existirem vértices apenas com entradas ou apenas com saídas.

Por inspeção, chegamos a uma matriz com os valores dos caminhos mais curtos para cada par vértice de excesso/defeito (ver Apêndice F).

Tendo em conta a tabela do conjunto dos caminhos ótimos de reposicionamento (ver Apêndice G), a solução sugere que se deve percorrer uma vez os caminhos $3 \rightarrow 5$, $6 \rightarrow 7$, $8 \rightarrow 4$, $8 \rightarrow 9$, $8 \rightarrow 12$, $10 \rightarrow 11$, $13 \rightarrow 2$, não necessariamente por esta ordem. A soma dos custos destes caminhos é igual a 131:

$$\begin{aligned}
&1 \times (3 \rightarrow 5) + 1 \times (6 \rightarrow 7) + 1 \times (8 \rightarrow 4) + 1 \times (8 \rightarrow 9) + 1 \times (8 \rightarrow 12) \\
&\quad + 1 \times (10 \rightarrow 11) + 1 \times (13 \rightarrow 2) = 6 + 16 + 30 + 24 + 31 + 3 + 21 \\
&\quad = 131
\end{aligned}$$

Capítulo 3

Conclusão

Culminada a elaboração do trabalho prático 2, atingindo-se o percurso ótimo respondendo ao objetivo proposto.

Ao nível de dificuldades sentidas, importa referir a complexidade em colocar no papel as restrições, visto que a sua idealização foi fácil, todavia transpor objetivamente foi um trabalho árduo.

Com o mesmo, foi permitida a consolidação de aprendizagens, nomeadamente o uso dos softwares LPSOLVE e RELAX4 que facilitou a realização dos cálculos necessários.

No ímpeto geral o desenvolvimento do trabalho decorreu como planeado, alcançando o objetivo esperado e a solução para o problema proposto.

Apêndice A

Modelo Programação Linear

```
/* Objective function */
min: 3 y12 + 3 y23 + 3 y27 + 6 y35 + 2 y43 + 6 y48 + 4 y54 +
2 y56 + 10 y61 + 8 y78 + 4 y79 + 4 y86 + 2 y910 + 12 y913 +
3 y1011 + 7 y118 + 2 y1112 + 5 y1210 + 3 y1213 + 4 y138 + 93;

/* Variable bounds */

y12 >= 0;
y23 >= 0;
y27 >= 0;
y35 >= 0;
y43 >= 0;
y48 >= 0;
y54 >= 0;
y56 >= 0;
y61 >= 0;
y78 >= 0;
y79 >= 0;
y86 >= 0;
y910 >= 0;
y913 >= 0;
y1011 >= 0;
y118 >= 0;
y1112 >= 0;
y1210 >= 0;
y1213 >= 0;
y138 >= 0;


y12 + 1 = y61 + 1;
y12 + 1 = y27 + y23 + 2;
y27 + 1 = y78 + y79 + 2;
y35 + 1 = y23 + y43 + 2;
y35 + 1  = y56 + y54 + 2;
y54 + 1  = y43 + y48 + 2;
y61 + 1  = y86 + y56 + 2;
y79 + 1  = y910 + y913 + 2;
y86 + 1  = y78 + y138 + y48 + y118 + 4;
y1011 + 1  = y910 + y1210 + 2;
```

```
y1011 + 1  = y118 + y1112 + 2;
y1112 + 1  = y1213 + y1210 + 2;
y138 + 1  = y913 + y1213 + 2;


int y12, y23, y27, y35, y43, y48, y54, y56, y61,
y78, y79, y86, y910, y913, y1011, y118, y1112,
y1210, y1213, y138;
```

Apêndice B

Ficheiro de input - Parte 1

```
13
20
1 2 3 100
2 3 3 100
2 7 3 100
3 5 6 100
4 3 2 100
4 8 6 100
5 4 4 100
5 6 2 100
6 1 10 100
7 8 8 100
7 9 4 100
8 6 4 100
9 10 2 100
9 13 12 100
10 11 3 100
11 8 7 100
11 12 2 100
12 10 5 100
12 13 3 100
13 8 4 100
0
-1
1
-1
-1
1
-1
3
-1
1
-1
-1
1
```

Apêndice C

Ficheiro de output - Parte 1

```
END OF READING
NUMBER OF NODES =          13 , NUMBER OF ARCS =          20
CONSTRUCT LINKED LISTS FOR THE PROBLEM
CALLING RELAX4 TO SOLVE THE PROBLEM
***********************************
TOTAL SOLUTION TIME =   1.17000003E-04  SECS.
TIME IN INITIALIZATION =   8.49999997E-05  SECS.
           1           2   5.0000000000000000
           2           3   1.0000000000000000
           2           7   3.0000000000000000
           3           5   2.0000000000000000
           5           4   1.0000000000000000
           6           1   5.0000000000000000
           7           9   2.0000000000000000
           8           6   4.0000000000000000
           9          10   1.0000000000000000
          10          11   2.0000000000000000
          11          12   1.0000000000000000
          13           8   1.0000000000000000
OPTIMAL COST =    131.00000000000000
NUMBER OF AUCTION/SHORTEST PATH ITERATIONS =          51
NUMBER OF ITERATIONS =           25
NUMBER OF MULTINODE ITERATIONS =           6
NUMBER OF MULTINODE ASCENT STEPS =           3
NUMBER OF REGULAR AUGMENTATIONS =           6
***********************************
```

Apêndice D

Ficheiro de input - Parte 2

```
13
35
3     2   21    100
3     4   10    100
3     5    6    100
3     7   24    100
3     9   28    100
3    11   33    100
3    12   35    100
6     2   13    100
6     4   26    100
6     5   22    100
6     7   16    100
6     9   20    100
6    11   25    100
6    12   27    100
8     2   17    100
8     4   30    100
8     5   26    100
8     7   20    100
8     9   24    100
8    11   29    100
8    12   31    100
10    2   27    100
10    4   40    100
10    5   36    100
10    7   30    100
10    9   34    100
10   11    3    100
10   12    5    100
13    2   21    100
13    4   34    100
13    5   30    100
13    7   24    100
13    9   28    100
13   11   33    100
13   12   35    100
0
-1
```

```
1
-1
-1
1
-1
3
-1
1
-1
-1
1
```

Apêndice E

Ficheiro de output - Parte 2

```
END OF READING
NUMBER OF NODES =          13 , NUMBER OF ARCS =          35
CONSTRUCT LINKED LISTS FOR THE PROBLEM
CALLING RELAX4 TO SOLVE THE PROBLEM
***********************************
TOTAL SOLUTION TIME =   1.28999993E-04  SECS.
TIME IN INITIALIZATION =   8.90000010E-05  SECS.
           3           5   1.0000000000000000
           6           7   1.0000000000000000
           8           4   1.0000000000000000
           8           9   1.0000000000000000
           8          12   1.0000000000000000
          10          11   1.0000000000000000
          13           2   1.0000000000000000
OPTIMAL COST =    131.00000000000000
NUMBER OF AUCTION/SHORTEST PATH ITERATIONS =          28
NUMBER OF ITERATIONS =          19
NUMBER OF MULTINODE ITERATIONS =           4
NUMBER OF MULTINODE ASCENT STEPS =           0
NUMBER OF REGULAR AUGMENTATIONS =           4
***********************************
```

Apêndice F

Tabela de custo de caminhos de reposicionamento

Exc. \Def.	2	4	5	7	9	11	12
3	21	10	6	24	28	33	35
6	13	26	22	16	20	25	27
8	17	30	26	20	24	29	31
10	27	40	36	30	34	3	5
13	21	34	30	24	28	33	35

Apêndice G

Tabela com o conjunto ótimo de caminhos de reposicionamento

Exc. \ Def.	2	4	5	7	9	11	12
3	3 -> 5 -> 6 - > 1 -> 2	3 -> 5 -> 4	3 -> 5	3 -> 5 -> 6 -> 1 -> 2 -> 7	3 -> 5 -> 6 -> 1 -> 2 -> 7 -> 9	3 -> 5 -> 6 -> 1 -> 2 -> 7 -> 9 -> 10 -> 11	3 -> 5 -> 6 -> 1 -> 2 -> 7 -> 9 -> 10 -> 11
6	6 -> 1 -> 2	6 -> 1 -> 2 -> 3 -> 5 -> 4	6 -> 1 -> 1 -> 3 -> 5	6 -> 1 -> 2 -> 7	6 -> 1 -> 2 -> 7 -> 9	6 -> 1 -> 2 -> 7 -> 9 -> 10 -> 11	6 -> 1 -> 2 -> 7 -> 9 -> 10 -> 11 -> 12
8	8 -> 6 -> 1 -> 2	8 -> 6 -> 1 -> 2 -> 3 -> 5 -> 4	8 -> 6 -> 1 -> 2 -> 3 -> 5	8 -> 6 -> 1 -> 2 -> 7	8 -> 6 -> 1 -> 2 -> 7 -> 9	8 -> 6 -> 1 -> 2 -> 7 -> 9 -> 10 -> 11	8 -> 6 -> 1 -> 2 -> 7 -> 9 -> 10 -> 11 -> 12
10	10 -> 11 -> 8 -> 6 -> 1 -> 2	10 -> 11 -> 8 -> 6 -> 1 -> 2 -> 3 -> 5 -> 4	10 -> 11 -> 8 -> 6 -> 1 -> 2 -> 3 -> 5	10 -> 11 -> 8 -> 6 -> 1 -> 2 -> 7	10 -> 11 -> 8 -> 6 -> 1 -> 2 -> 7 -> 9	10 -> 11	10 -> 11 -> 12
13	13 -> 8 -> 6 -> 1 -> 2	13 -> 8 -> 6 -> 1 -> 2 -> 3 -> 5 -> 4	13 -> 8 -> 6 -> 1 -> 2 -> 3 -> 5	13 -> 8 -> 6 -> 1 -> 2 -> 7	13 -> 8 -> 6 -> 1 -> 2 -> 7 -> 9	13 -> 8 -> 6 -> 1 -> 2 -> 7 -> 9 -> 10 -> 11	13 -> 8 -> 6 -> 1 -> 2 -> 7 -> 9 -> 10 -> 11 -> 12